La revue francophone indépendante pour les utilisateurs des Apple][+, //e, //e+, //c, IIGS™ et Macintosh™

Pom'S

SOFTWARE

Exclusivité Pom's : les TimeOut en 'vf' !

M 2366 - 38 - 45,00 F
3792366045005 00380
ISSN 0294-6068

Apple Expo 88

Le monde se rejoint à la Villette.

Cette année, le succès d'Apple dans l'entreprise devrait ajouter à Apple Expo une saveur toute particulière. Le monde du travail rejoint en effet celui des passionnés. Et chacun se retrouve sur une idée : c'est avec des machines puissantes, mais simples d'accès, que l'on donnera à l'Homme les moyens d'exprimer le meilleur de lui-même, les ressources à l'intérieur.

Et c'est ainsi tout un point de vue sur la productivité qui s'impose dans la Grande Halle de la Villette.

Avec 150 exposants : partenaires d'Apple, créateurs et éditeurs de logiciels, constructeurs de périphériques, organismes de formation et éditeurs de presse. Mais aussi des conférences, et bien sûr la « Ville Apple ».

Pour que demain soit vraiment un autre jour. De plus en plus de performances? Certes, mais qui permettent à chacun de développer sa propre richesse.

Et là, tout le monde peut se rejoindre. Ouf.

Apple

Du mercredi 28 septembre au samedi 1er octobre. De 10 à 19 h. Métro Porte de Pantin. La Grande Halle, la Villette. Pour le calendrier des conférences ou la grande halle d'autres informations : 3614 code : APPLE.

C.L.M. & BBDO. Apple, le logo Apple, sont des marques déposées de Apple Computer Inc.

Numéro 38
septembre-octobre 1988

Éditorial — Hervé Thiriez — Page 7

Interview : Randy Brandt — Page 8

Ensoniq : Marc Bavant
Travaux pratiques (2) — Page 11

Les disquettes 'Polices' — Page 16

Da.Mover — Frédéric Rozay — Page 17

Les TimeOut en VF — Dimitri Geystor — Page 23

Le calcul des Premiers
Prime.Factor — Pascal Cantot — Page 27

Applesoft & AppleWorks — Dimitri Geystor — Page 31

AW sur 2 colonnes — A.-P. Gaspar — Page 37

AppleWorks new look, suite — Éric Weyland — Page 39

PR#6 — Jean-Luc Bazanegue — Page 41

Puzzle — Marc Girati — Page 44

Macintosh : les news — Ariel Sebban — Page 50

La disquette Mac 38 — Page 52

Apple // for ever — Jean-Yves Bourdin — Page 53

Courrier des Lecteurs — Page 66

Sommaire des Pom's — Page 68
Petites annonces — Page 74
Mots croisés

Les annonceurs ; Apple : p 2 ; Bréjoux A.E. : p 55 ; P. Carlier : p 13 ; Logma S.A. : p 73 ; Ubi Soft : p 6

Éditions MEV - 12, rue d'Anjou - 78000 Versailles. Tél. : (1) 39 51 24 43. Directeur de la publication : Hervé Thiriez

Imprimé en France. N° d'impression 784509 — Dépôt légal : septembre 1988.

Certaines nuits ne portent pas conseil...

Lorsqu'au petit matin le problème demeure, il reste la carte Joker. Elle donne un accès privilégié à notre Hotline téléphonique. Gratuite, la carte Joker accompagne chaque abonnement ; un nouvel atout pour les programmes Pom's.

Copy II Plus 8.3

Apple //

- Utilitaire complet pour la gestion des disques et fichiers DOS/ProDOS, y compris conversion automatique
- Utilisation de toute la mémoire disponible (un 800Ko en une passe par exemple)
- Récupération de fichiers effacés
- Formatage de disques
- Classement alphabétique des fichiers
- Sauvegarde de la plupart des logiciels…

Copy II Plus 8.3 sur disquette 5,25" et 3,5",
avec documentation française : 500,00 F TTC
Abonnés (carte Joker) : 450,00 F TTC
Port : 20,00 F

Copy II 7.1

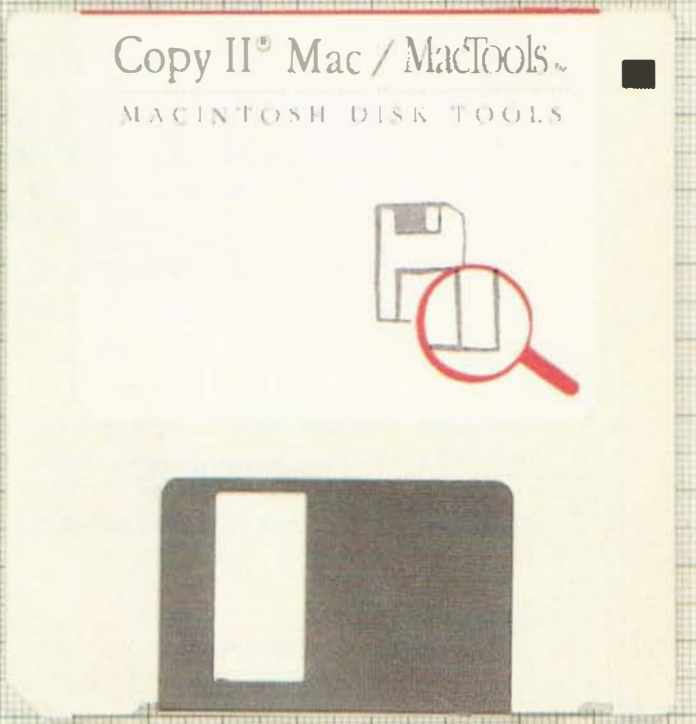
Macintosh

- Copie de sauvegarde de la plupart des logiciels
- Copie de la plupart des logiciels sur disque dur
- Récupération de disques endommagés
- Récupération de fichiers effacés même sur disque dur

…

Copy II est livré avec MacTools

Copy II Mac 7.1, MacTools,
avec documentation française : 500,00 F TTC
Abonnés (carte Joker) : 450,00 F TTC
Port : 20,00 F



MANUELS EN FRANÇAIS

Éditions MEV/Pom's – 12, rue d'Anjou – 78000 Versailles
(1) 39 51 24 43. Commande par Minitel au (1) 39 53 04 40

UN CHALLENGE DE HAUT NIVEAU

Du fait de son ouverture sur les marchés extérieurs et de son expansion internationale, **UBI SOFT** recherche :

- DES PROGRAMMEURS EXCEPTIONNELS en C, Assembleur et Turbo Pascal

- ayant déja réalisé leur programme et voulant être édités ;
- pouvant assurer la conversion de logiciels sur différents formats ;
- voulant voir leur programme distribué dans tous les pays du monde.

- DES GRAPHISTES DE TALENT

- qui pensent pouvoir égaler et même surpasser les graphismes de Zombi, Iron Lord ou Rocket Ranger ;
- qui ont une imagination débordante.

- DES MUSICIENS DE GÉNIE

- qui ont envie de mettre leur talent au service de la création ;
- pour qui la musique sur micro-ordinateur n'a plus de secret.

VOUS TRAVAILLEZ

sur Macintosh, Apple II, Apple II GS, et vous êtes passionnés par ces machines

ou vous connaissez l'Amstrad, le PC, M.S.X., Spectrum, Commodore 64, ST, Amiga, Nintendo, M.S.X. II.

Un futur plein d'avenir peut s'ouvrir à vous.
Aujourd'hui, téléphonez ou écrivez à **UBI SOFT.**
Demain vous serez peut-être choisis pour faire partie d'une équipe de gagnants.

POM'S

NOM..... Prénom .....

Adresse .....

.....

..... Tél. .....

Langage ..... Machine .....

A retourner à **UBI SOFT** / Christine QUÉMARD
1, voie Félix Eboué - 94021 CRETEIL CEDEX - Tél. (1) 48 98 99 00

Ont collaboré à ce numéro
Marc Bavant – Jean-Luc Bazanegue
Jean-Yves Bourdin - Pascal Cantot
A.-P. Gaspar - Dimitri Geystor
Marc Girati - Olivier Herz
Gérard Michel - Christian Piard
Joëlle Piard - Frédéric Rozay
Hervé Thiriez - Éric Weyland

Directeur de la publication
rédacteur en chef
Hervé Thiriez

Rédacteurs
Jean-Luc Bazanegue - Christian Piard

Siège social
Éditions MEV – 12, rue d'Anjou
78000 Versailles – ☎ (1) 39 51 24 43
Serveur Minitel (1) 35 53 04 40

Publicité
Éditions MEV

Diffusion
N.M.P.P.

Impression
Berger-Levrault
18, rue des Glacis
54000 Nancy
☎ 83.35.61.44

Photos
CP & JLB

Photogravure
Graphotec
21, chemin de la Tour
92350 Le Plessis-Robinson
☎ (1) 46 30 44 49

Pom's est une revue indépendante non rattachée à Apple Computer, Inc. ni à Apple Computer France S.A.R.L.
Apple, le logo Apple, Mac et le logo Macintosh sont des marques déposées d'Apple Computer, Inc.
IBM est une marque déposée de International Business Machine.
PC et AT sont des marques déposées de la Société IBM.

©Éditions MEV 1988
Toute reproduction intégrale ou partielle, effectuée par quelque procédé que ce soit, sans l'accord écrit d'Éditions MEV, constitue une contrefaçon.
Loi du 11 mars 1957, articles 425 et suivants du Code Pénal.
Droits de traduction, de reproduction et d'adaptation réservés pour tous pays.

Éditorial

L'événement TimeOut

Ce n'est pas une nouveauté, c'est véritablementl'événement de cette rentrée pour le monde Apple //. Il n'y avait pas d'espoir de voir adapté à notre AppleWorks 1.4 le formidable potentiel des TimeOut : comme nous le disions dans le précédent numéro, nous avons choisi de retrousser nos manches parce que, pour Pom's, la diffusion de logiciels essentiels n'est pas comprise comme une simple affaire d'achat/vente.

Nous vous proposons aujourd'hui les premiers fruits de ce travail d'équipe qui nous a conduit des négociations préalables avec les détenteurs de droits américains jusqu'à ces quatre premiers TimeOut, travail de modification et d'adaptation des softs, de francisation des écrans, de traduction des manuels.

De nouveaux fruits, de nouvelles surprises dans les prochains numéros. Bientôt, dans ces colonnes, l'art et la manière d'écrire soi-même son application TimeOut avec exemple et sources.

Virus

Pom's ne tombe pas dans la viromania médiatique mais lorsque l'un des plus grands éditeurs de softs pour Macintosh constate *in extremis* que des disquettes à diffuser ne sont pas saines... De façon accessoire, nous vous offrons sur la disquette de ce numéro six outils Mac qui protègent contre les effets de ces routines malignes.
Comme d'une vaccination ou d'un médicament, n'en attendez pas une universalité : de nouveaux virus mutants et résistants guettent !

Apple Expo

Notre rendez-vous annuel vous permettra cette année de rencontrer quelques auteurs de la revue (voir page 67). Vous êtes auteurs ou futurs auteurs ? Ne manquez pas de vous présenter et de nous parler de vos projets : c'est par l'écoute attentive de ses lecteurs que Pom's évolue de numéro en numéro.

Hervé Thiriez

À gauche Randy Brandt accompagné de B. Lissner, auteur d'AppleWorks

Interview : Randy Brandt

Le nom de Randy Brandt, des Beagle Brothers, est déjà bien connu de nos lecteurs. À l'occasion de la sortie des premières applications TimeOut en version française, Pom's lui a posé, pêle-mêle, quelques questions sur son travail, ses projets, le métier de programmeur, les copies-pirates, l'avenir des Apple //, etc.

Pom's — Avez-vous déjà écrit de nombreux programmes, et lesquels ?

Randy Brandt — Je vous en donne une liste dans l'ordre chronologique : Big-U, MacroWorks, SuperMacroWorks, PathFinder, TimeOut UltraMacros, PatchMania, TimeOut MacroTools et TimeOut PowerPack Mais j'ai aussi contribué aux disques suivants : Extra K, ProByter, Minipix, Late Nite Patches, TimeOut DeskTools II et TimeOut SpreadTool.

Pom's — De tous ceux-ci, en est-il un dont vous êtes particulièrement fier ?

Randy Brandt — C'est sans doute TimeOut UltraMacros : j'ai réussi, en très peu de place, à installer un arsenal de commandes d'une puissance étonnante, assorti en même temps d'une très grande souplesse d'emploi pour l'utilisateur. Et j'avoue avoir aussi un faible pour PathFinder : il fonctionnait deux jours après que j'en ai eu l'idée...

Pom's — Nos lecteurs de Pom's aimeraient en savoir un peu plus sur vous. Est-il indiscret de vous demander de vous présenter ?

Randy Brandt — Je suis né au Canada, en 1960. Plus précisément au Manitoba — la neige et les grands espaces. Et puis, à l'âge de 19 ans, je me suis transporté loin vers le sud : en Californie, à plus de 3 000 kilomètres. J'y ai suivi des études universitaires au Christian Heritage College, avec comme option principale... la biologie. C'est là que j'ai rencontré ma femme Joanna, c'est là aussi que j'ai commencé à programmer sur un Apple II+. Il n'y avait pas, je précise, de cours de programmation dans mon établissement (j'ai d'ailleurs terminé avec un diplôme de littérature anglaise) ; j'ai appris en autodidacte, en piochant dans Nibble et Call-Apple.

Une fois mes diplômes en poche, j'ai épousé Joanna et me suis établi aux États-Unis. Deux ans plus tard (en 1985), les Beagle Bros m'ont engagé pour travailler à la documentation d'Extra K et de ProByter. Cette même année j'ai écrit Big-U (un utilitaire sous ProDos), puis vinrent les droits d'auteur de MacroWorks. C'est depuis cette époque que, ma foi, la vie me sourit.

Pom's — C'est-à-dire ? Estimez-vous mieux gagner votre vie que, mettons, un médecin, un journaliste, un avocat ? Et pour y arriver, combien d'heures travaillez- vous au total par semaine ?

Randy Brandt — Un programmeur peut bien gagner sa vie à condition d'avoir sorti un 'best seller', ou même plusieurs. Soyons francs : peu d'entre nous y parviennent. Mes 'améliorations' à AppleWorks ont été d'un excellent rapport, mais de là à dire que je réussis aussi bien qu'un avocat ou un médecin... Ma femme et moi avons des goûts modestes ; nous sommes heureux dans notre petite maison : nous avons notre chambre, nos deux petites filles (Heather — 3 ans — et Erica — 1 an —) ont la leur, et... mes ordinateurs occupent la troisième. Nous nous estimons privilégiés d'avoir notre maison, mais nous la payons à tempérament, comme tout le monde.

Je travaille environ 60 heures par semaine. Je commence rarement avant 9h30 du matin, et je continue en général au-delà de minuit. Comme je fais des pauses pour langer Erica, ma femme ne me tient pas trop rigueur

des couchers tardifs. De toutes façons, elle m'a toujours sous la main !

Le vendredi est le jour des Beagle Bros : nous nous retrouvons tous au 'bureau', autour d'un lunch de travail : le fondateur des Beagle Bert Kersey, le président de la société Mark Simonsen, les programmeurs Alan Bird, Rob Renstrom, Dan Verkade, et moi-même. Nous échangeons des idées, montrons nos dernières 'œuvres' ; mais la discussion est tout aussi animée qu'il s'agisse d'ordinateurs, de sport, de politique ou de n'importe quel autre sujet.

Le dimanche est consacré au culte (église Évangélique protestante), aux amis et à la détente. Il m'arrive parfois, les dimanches après-midi, de faire mon courrier pendant que les 'dames de la maison' fontla sieste. Toutefois, j'essaie de m'astreindre, autant que possible, à accorder aussi à mes ordinateurs leur repos dominical.

Mais comme la programmation est un exercice plus mental que physique, je n'arrive jamais à y échapper totalement. Il faut croire que ça 'travaille dans ma tête' même quand je dors, car il m'arrive de me coucher après avoir vainement tenté de résoudre un problème, et d'aller droit à la solution le lendemain matin.

Pom's — Et les loisirs ?

Randy Brandt — Une fois les ordinateurs éteints ? Il ne leur reste pas beaucoup de place. Par contre, j'ai une télé dans mon bureau, et j'aime bien suivre les sports tout en programmant. J'aime aussi la lecture (fiction et réalité), m'instruire en matière politique et religieuse, jouer aux échecs, au basket et autres sports. J'adore aussi jouer avec mes deux petites filles.

Pom's — Quels conseils donneriez-vous à un jeune qui rêve de devenir le Randy Brandt de l'an 2010 ?

Randy Brandt — Si un écolier me demandait comment on devient un programmeur (qui réussit !), je lui conseillerais, avant toute chose, de suivre des cours de dactylographie, de géométrie et de maths. Tout logiciel représente un ensemble de problèmes conceptuels et logiques : des connaissances d'algèbre et de géométrie sont particulièrement utiles. De plus, on ne programme pas bien si l'on ne sait pas bien taper à la machine : il faut donc absolument en passer par là.

Bien que je ne pratique que le Basic et l'Assembleur, je recommanderais aux programmeurs à venir d'apprendre le langage C. C'est le langage du futur, et je compte d'ailleurs m'y mettre moi-même. Mes amis Mark, Alan et Rob ont tous les trois appris la programmation à l'université, et ça leur donne un avantage sur moi. Quant à moi, j'ai appris la programmation sur le peu de temps de libre que me laissaient mes autres études, l'activisme étudiant (nous appelons cela 'Student government'), les articles que j'écrivais dans le journal universitaire, etc., etc. Bref, je me suis formé en lisant des revues. Si c'est votre voie, épluchez tous les programmes de Pom's : il ne suffit pas de les recopier, il faut s'en imprégner si vous voulez programmer à votre tour.

Enfin, il faut que programmer soit pour vous un PLAISIR. Sinon, n'insistez pas, allez chercher fortune ailleurs. À ma connaissance, tous les programmeurs qui ont réussi adorent leur métier. J'aime à dire en boutade que je n'ai pas de métier, juste un violon d'Ingres à plein temps !

Pom's — Revenons aux Beagle. Pour le moment, l'essentiel de leur production — nous voulons parler de la nouvelle série TimeOut — se situe dans le sillage d'AppleWorks. Ne craignez-vous pas de mettre 'tous vos œufs dans le même panier' ? Bientôt, Claris va sortir le nouvel AppleWorks GS. N'allez-vous pas être débordés ?

Randy Brandt — Non, pas du tout. AppleWorks GS sera très lent par rapport à AppleWorks, et fort gourmand en mémoire (1,25 Méga pour le moins). Il ne devrait pas 'faire de l'ombre' à la série TimeOut. Beaucoup de ceux qui possèdent un GS hésiteront à sortir quelques centaines de dollars — ou milliers de francs — uniquement pour faire tourner un programme affligé d'une interface graphique poussive. D'ailleurs même les //c et //e continuent à bien se vendre (du moins aux États-Unis), et il y a un grand nombre d'utilisateurs d'AppleWorks qui ne sont pas encore passés à TimeOut, et qui y viendront.

Je prédis, personnellement, que la majorité de ceux qui achèteront l'AppleWorks GS continueront à faire l'essentiel de leur travail avec AppleWorks et TimeOut, et ne passeront à l'AppleWorks GS que pour des applications spéciales, pour des mises en page élaborées par exemple. Il est bon que Claris ait pris une initiative favorable à l'Apple //, et je m'en réjouis ; mais pour abattre du boulot, vite fait bien fait, le bon vieil AppleWorks est loin d'avoir dit son dernier mot. D'ailleurs, à l'AppleFest du mois de mai dernier, tant Bob Lissner (l'auteur d'AppleWorks) que Claris (son distributeur) ont annoncé une mise à jour de la dernière version (anglaise) d'AppleWorks.

Pom's — Programmez-vous seul, ou en équipe ? Comment coordonnez-vous votre travail ?

Randy Brandt — Tous les programmeurs de chez Beagle travaillent généralement seuls, dans la mesure où ils ne mettent pas leurs fichiers-sources en commun. Mais on peut dire aussi qu'ils travaillent en équipe car ils essaient mutuellement leurs programmes, se font des suggestions, ou même échangent des idées à développer. Je vous ai déjà parlé de nos réunions du vendredi : c'est au cours de ces lunchs que naît l'émulation. Nous sommes, en quelque sorte, une grande famille : nos épouses et nos enfants se connaissent, et nous fêtons ensemble nos divers anniversaires, en toute amitié. Il n'y a pas de cloison étanche entre le travail et le reste de l'existence. Nos âges ? En moyenne, autour de trente, un peu moins peut-être. Matt Reimer est l'exception : il avait 16 ans quand il a commencé chez les Beagle Bros, il en a 19 aujourd'hui.

Matt a travaillé sur les 'Beagle Screens'. Il écrit actuellement un programme de dessin pour IIGS nommé 'MiniPaint', qui sera distribué par JEM SoftWare dès l'hiver prochain pour un prix de 25 dollars. Il en sortira une version plus puissante d'ici une année sans doute, qui sera publié

cette fois par les Beagle Bros. J'ai fait la connaissance de Matt alors que je faisais un remplacement en tant que prof de chimie à son école, à l'époque où il faisait ses premières armes chez les Beagle Bros. Et c'est pas hasard que nous avons découvert que nos grands-pères avaient été de bons amis, il y a 50 ans, au Canada !

***Pom's** — Quels sont vos langages de programmation favoris ? Quels outils de programmation utilisez-vous ? Sur quelle(s) machine(s) ?*

Randy Brandt — Je n'écris qu'en assembleur 6502, à l'exception des programmes d'installation qui sont en Basic Applesoft. Pardon, j'utilise aussi un troisième langage de programmation : c'est UltraMacros sur AppleWorks !

Tout récemment, je suis passé au Macintosh Programmer's Workshop sur Macintosh SE, et à son assembleur IIGS. Vu sa puissance, je crois que je finirai par l'utiliser pour tout mon travail futur. Mais la totalité de mes programmes actuels ont été écrits avec Merlin Pro, d'abord sur un //c, puis sur un IIGS.

Voici une liste de mon matériel :

- Un Apple IIGS avec 1,75 Mégas (GS RAM d'Applied Engineering)
- 1 carte RamFactor (1 Méga) avec batterie autonome (Merlin, AppleWorks, etc.)
- 1 lecteur 3,5"
- 1 lecteur 5"1/4
- 1 modem 1200 bauds
- 1 écran //e vert Apple //e (pour la programmation)
- 1 écran couleur IIGS (pour épater les amis, et pour les logiciels scolaires "Sticky Bear" de ma fille Heather)
- 1 imprimante ImageWriter II
- 1 Apple //c
- 1 Macintosh SE avec disque dur interne 20 Méga.

(Mes amis Roger et Tina Wunker, qui font toutes les expéditions des logiciels JEM Software, gèrent la liste des clients avec le //c et la Base de données d'AppleWorks).

Jusqu'à présent, je n'ai employé que Word de MicroSoft, et l'MPW sur le Mac.

Pour gérer toutes mes fiches, faire mon courrier, etc. j'emploie AppleWorks sur le IIGS. Je réserve Word à des travaux spéciaux, que je souhaite imprimer sur la LaserWriter des Beagle. Nous faisons aussi la mise en page de nos manuels avec Word, bien qu'ils soient d'abord écrits sur AppleWorks, puis transférés sur Mac.

***Pom's** — Vous avez toujours eu pour politique (comme Pom's) de vendre des programmes non protégés, malgré le risque de les voir piratés. Et pourtant, malgré (ou à cause de ?) leur prix très raisonnable, ils continuent à bien se vendre. Qu'avez-vous à dire à ce propos ?*

Randy Brandt — Ce n'est pas une 'protection' qui découragera les déplombeurs ; je dirai même qu'ils y voient un défi. À mon avis, face à un bon logiciel non protégé, à un prix abordable, une personne honnête reste honnête, alors que le pirate endurci n'y trouve aucune stimulation. Quant au 'gagne-petit' décidé à se procurer des logiciels gratis, il y parvient de toutes façons, protection ou pas. C'est regrettable, mais cela fait partie des désagréments du métier. Il serait injuste de vouloir imposer, pour une protection illusoire, des désagréments à la majorité des acheteurs honnêtes. Il est très difficile, pour ne pas dire impossible, d'utiliser des disques RAM et des disques durs avec des logiciels protégés. Notre politique a toujours été de proposer des logiciels d'accès simple et convivial, et notre clientèle nous en sait gré.

***Pom's** — Cette clientèle vous écrit-elle beaucoup ? Comment arrivez-vous à lui répondre et à programmer quand-même ?*

Randy Brandt — Oui, le courrier ne manque pas : je lui consacre plusieurs heures par semaine ! UltraMacros m'apporte des consolations : la correspondance me prend du temps, mais m'amène souvent de bonnes idées. Je crois que mes correspondants apprécient mes propres suggestions à leurs questions

***Pom's** — Et Apple, dans tout ça ? Que pensez-vous du IIGS ? Quand on a un //e, faut-il le vendre, ou le mettre à niveau ? Quel est l'avenir de l'Apple // en général ?*

Randy Brandt — L'avenir du IIGS se présente bien. Il est de plus en plus question d'un modèle amélioré, plus rapide, avec une meilleure résolution verticale et d'autres 'plus'. Les premières innovations ne viendront peut-être pas d'Apple. Déjà, des sociétés comme Applied Engineering travaillent sur des accélérateurs et des cartes mémoire perfectionnées.

Apple soutient la famille des Apple // (du moins aux U.S.A.) en proposant un //c équipé d'un lecteur interne 3,5". Si Apple pensait que la famille des // était morte et enterrée, il ne se donnerait pas la peine de sortir un nouveau //c.

Vu le nombre d'Apple // couramment utilisés dans le monde, je crois que même si Apple arrêtait d'en fabriquer, ils auraient encore un bel avenir devant eux. Connaissez-vous une autre machine qui compte un tel nombre d'utilisateurs dévoués ? Il appartient à des publications comme Pom's, Nibble, Open Apple et d'autres de faire circuler l'information, maintenir un lien entre les membres de cette immense communauté, de ne pas laisser s'éteindre l'enthousiasme.

Quant à moi, j'ai été heureux de cette occasion de partager un peu de mon univers personnel avec les lecteurs de Pom's. J'applaudis aux efforts de votre revue qui a su garder à l'Apple // une place de choix dans un pays quelque peu saisi par la 'fièvre de Mac', et je vous envoie à tous un très amical salut !

2/09/88

Disquettes DPA 1 à 7

Vous trouverez la description de nos sept premières disquettes Apple // Shareware dans le numéro 37 de Pom's ou sur notre serveur Minitel au : (1) 39 53 04 40.

Ensoniq : Travaux pratiques

Marc Bavant

La première partie de cet article, publiée dans le numéro 37 de Pom's, nous a permis de faire connaissance avec la 'théorie' de l'Ensoniq ; passons donc maintenant à la pratique...

Étalonnage du convertisseur

Nous savons peu de choses sur ce convertisseur sinon qu'il accepte des tensions comprises entre 0 et 2,5 V. Construisons-nous un petit programme qui se charge d'interroger le convertisseur et de stocker 256 résultats successifs.

```
*!
!800:LDA £13
! STA C03C
! LDA £E2
! STA C03E
! LDX £0
!80C:LDA C03D
! DEC 0      ; histoire de passer 5
cycles
! DEC 0
! DEC 0
! DEC 0
! DEC 0
! DEC 0
! STA 2000,X
! INX
! BNE 80C
! RTS
!
*
```

Pour fonctionner comme souhaité, ce programme doit être lancé en vitesse lente (sélectionnez *vitesse normale* dans le tableau de bord, revenez sous Basic et faites un Reset pour être plus sûr). En vitesse rapide, il faudrait ajouter des délais pour atteindre les 32 microsecondes indispensables (faites-le à titre d'exercice).

Vérifier à l'aide d'un voltmètre que le circuit ne délivre aucune tension entre les broches 1 et 2 (si c'était le cas, vous vous seriez trompé de broches !) et court-circuitez-les. La valeur trouvée par le convertisseur est alors 0. Si vous vous en sentez capables, bricolez une alimentation variable avec une pile et un potentiomètre. Reliez la masse au pôle *moins* de la pile et appliquez prudemment et successivement des tensions de 0 à 2,5 V de manière à étalonner le convertisseur. Pour ma part, aux erreurs de mesure près je trouve :

tension (V)	valeur numérique
0.0	00
0.2	16
0.4	2C
0.6	42
0.8	58
1.0	6E
1.2	80
1.4	96
1.6	AA
1.8	BE
2.0	D2
2.2	E8
2.4	FC

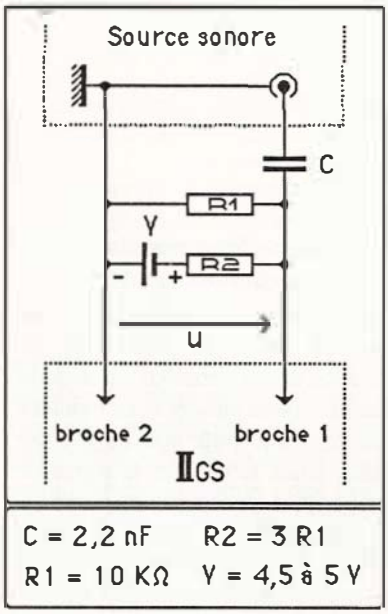
Ces résultats montrent de manière flagrante que le convertisseur est linéaire sur la totalité de son domaine, avec une pente constante d'environ 110/V.

Cette gamme de tensions admissibles va nous poser un petit problème car la plupart des équipements de reproduction musicale (postes de radio, magnétophones...) fournissent un signal de valeur moyenne nulle, signal qu'il faudra donc polariser autour de 1,2 V si on veut l'utiliser en entrée du convertisseur. Et ceux qui fourniraient éventuellement un signal polarisé n'ont aucune raison d'avoir la tension moyenne que nous recherchons. Un simple couplage par condensateurs et un pont diviseur tel que celui décrit ci-après feront l'affaire dans tous les cas.

Attention : Mesurez la tension U avant de relier le DOC au montage, U = polarisation continue + signal et vérifier impérativement :

0 <= U <= 2,5 V

Ce montage ne protège absolument pas le DOC des surtensions ni d'une erreur de polarité dans le branchement de la pile. Si vous ne savez pas distinguer le pôle '+' d'une pile de son pôle '–' réclamez des dommages-intérêts au Ministère de l'Éducation Nationale.



Adaptateur d'entrée du convertisseur analogique ➡ numérique

Écoute directe du son numérisé et écrêté

Ce paragraphe va certainement rappeler des souvenirs à ceux qui avaient utilisé l'entrée cassette de leur Apple][+ ou //e pour numériser des messages qu'on pouvait lui faire restituer d'une voix nasillarde, mais parfaitement intelligible et naturelle, à tout moment. La technique est ici la même : on ne regarde que le signe du signal d'entrée (par rapport à sa valeur moyenne) et on actionne un déplacement de la membrane du haut-parleur à chaque changement de signe.

actionne un déplacement de la membrane du haut-parleur à chaque changement de signe.

```
*!
!800:LDA £13
! STA C03C
! LDA £E2
! STA C03E

!80A:LDX C03D   ; 4 cycles
! JSR 830       ; 12 + n*2 (n =
  nombre de NOP)
! TXA           ; 2
! EOR 0         ; 3
! BPL 80A       ; 3

! STX 0
! STA C030      ; haut-parleur
! BIT C061      ; bouton ou Ó
! BPL 80A
! RTS

!830:NOP
! NOP
! NOP
! NOP
! RTS
```

Une fois encore, les temps sont ajustés pour un fonctionnement à 1 MHz.

Testons tout cela. Choisissez une station de radio, de préférence en modulation de fréquence en raison de la meilleure qualité du son, et au moment des informations pour pouvoir évaluer l'intelligibilité. Récupérez le signal de la prise casque, par exemple grâce à un jack monophonique avec un éventuel adaptateur stéréo-mono, et enfin injectez ce signal à l'entrée du montage polarisateur après avoir pris soin de baisser le réglage de volume du poste. Il est prudent de garder un voltmètre en parallèle sur l'entrée du DOC pour surveiller les tensions appliquées.

Lancez le programme ci-dessus, ajustez le volume juste au-dessus du seuil limite, ne lésinez pas sur les aigus et appréciez la qualité de la parole écrêtée c'est-à-dire, dans notre cas, numérisée avec un seul bit par échantillon. Si tout fonctionne correctement, la parole doit être clairement intelligible et ressembler à ce qui peut sortir d'un walkie-talkie, d'un poste CB ou autres gadgets pas précisément renommés pour leurs qualités musicales. Si ça ne marche... pas cherchez l'erreur !

L'avantage de ce type de numérisation est bien entendu la compacité du codage qu'il est possible d'atteindre (pour mémoire, les systèmes similaires sur Apple // permettent de stocker une petite dizaine de secondes de parole dans 16Ko, mais cela dépend beaucoup de la qualité du son). La fréquence d'échantillonnage est toutefois un peu faible sur le DOC pour ce genre de technique (l'entrée cassette était mieux adaptée aux échantillonnages rapides sur un bit).

Quand vous en avez assez, frappez la touche Ó ou faites reset.

Numérisation et stockage du son dans la RAM privative

L'expérience précédente nous aura surtout permis de vérifier que notre montage fonctionne correctement. Nous allons maintenant préparer l'étape suivante en stockant les valeurs numérisées dans la RAM privative du DOC. On peut noter au passage qu'il est dommage que le processeur soit obligé d'intervenir pour cette opération qu'on aimerait bien voir le DOC effectuer seul. Le programme qui suit est une copie du précédent légèrement compliquée par la valse des registres.

```
*
*!800:LDA £00
! STA C03E
! STA C03F

(acquérir l'échantillon précédent et
relancer la numérisation)

!808:LDA C03E     ; 4 + k cycles
! PHA             ; 3
! LDA £13         ; 2
! STA C03C        ; 4 + k
! LDA £E2         ; 2
! STA C03E        ; 4 + k
! LDX C03D        ; 4 + k
! BNE 81C         ; 3 (remplacer les
  valeurs $00 par $01)
! INX             ; 2 - 1

(insérer l'échantillon dans la RAM
privative)

!81C:LDA £73      ; 2
! STA C03C        ; 4 + k
! PLA             ; 4
! STA C03E        ; 4 + k
! STX C03D        ; 4 + k

(continuer jusqu'à 64 Ko échantillons)

! LDA C03E        ; 4 + k
! BEQ 831         ; 3
! NOP             ; 2 - 1
! JMP 834         ; 3 (astuce pour
  équilibrer les temps de cycle)
!831:LDA C03F     ; 4 - 1 + k
!834:BNE 808      ; 3
! RTS
```

Lancez ce programme par 800G, le prompteur du moniteur réapparaîtra après un court instant quand toute la RAM privative aura été remplie.

Remarque : la valeur 0, qui correspond à une tension rigoureusement nulle, a la mauvaise propriété d'être également un ordre d'arrêt pour les oscillateurs qui essaient de la produire, d'où notre INX.

Remarque 2 : on a procédé à une astuce pour éviter que la durée de la période d'échantillonnage ne diffère selon la branche empruntée par le programme après le test de début de page. Ceci n'a sans doute pas grand effet sur la qualité du son et présente l'inconvénient d'allonger cette période dans tous les cas.

Exercice 5.1 : on arrive à une période d'échantillonnage de 59 ou 60 cycles si je ne me trompe pas, ce qui est un peu trop pour un fonctionnement du microprocesseur à 1 MHz et pas assez en principe pour 2,8 MHz. Néanmoins, on verra que grâce au ralentissement dû aux accès à la mémoire lente, ce programme fonctionne également en vitesse "rapide". En supposant que l'accès à la mémoire lente se fait exactement à 1 MHz, calculez le nombre équivalent k de cycles à 2,8 MHz "ajoutés" aux instructions STore et LoaD. (réponse : k=11 environ)

Exercice 5.2 : calculez la fréquence d'échantillonnage correspondant à ce programme tournant à 1 MHz. Calculez le débit binaire moyen de transfert dans la RAM des données numérisées. Calculez le temps de remplissage de la RAM.

Mesurez-le et comparez.

Exercice 5.3 : mesurez le temps de remplissage pour l'horloge à 2,8 MHz. Déduisez-en la période et la fréquence d'échantillonnage. Conclusion.

Nous pouvons lancer ce programme mais, à ce stade, nous n'aurons pas d'assurance sonore que tout s'est bien passé. Contentons-nous donc d'observer une page de la RAM privative et de vérifier qu'elle contient

des données plausibles.

```
*!
!840:LDX £0
! JSL FF40A8        ;ReadRam
!846:STA 2000,X
! JSL FF40AF        ;ReadNxt
! INX
! BNE 846
! RTS
!
* 840G 2000.20FF
```

Remarque : comme nous exécutons ce programme en mode 8 bits, l'octet de poids fort du registre X n'est pas positionné par nous, on observe donc bien une page de la RAM mais on ne sait pas laquelle *a priori*.

6. Les registres d'oscillateur

Voici l'heure du plat de résistance ! Les lecteurs pressés peuvent sauter directement aux expériences mais il ne comprendront pas tout. Il ne sera pas question ici d'entrer dans tous les détails, mais seulement d'essayer d'éclairer la documentation développeurs sans la paraphraser. La lecture intégrale de cette documentation (Ensoniq DOC ERS, 11 pages) n'est pas nécessaire pour comprendre ce chapitre, mais je la recommande fortement à ceux que le sujet intéresse.

Généralités sur les oscillateurs

Le DOC est un processeur synchronisé grâce à une horloge à 7,14 MHz (je ne garantis pas cette valeur sur laquelle la documentation semble hésitante). Le cycle de traitement interne du DOC est 8 fois plus long que le cycle d'horloge. Pendant chaque cycle de traitement, le DOC 'donne la parole' à un oscillateur, par ordre de numéro croissant et jusqu'à concurrence du nombre d'oscillateurs autorisés (si quatre oscillateurs sont autorisés, le DOC donnera successivement la parole au 0, au 1, au 2, au 3, au 0, etc.). Il consacre ensuite deux cycles au rafraîchissement de la RAM. La durée qui sépare deux interrogations successives du même oscillateur dépend donc du nombre d'oscillateurs autorisés. Par commodité, on lui donnera le nom de période d'échantillonnage du DOC (ne pas confondre cet échantillonnage de sortie avec l'échantillonnage de 32 microsecondes réalisé par le convertisseur d'entrée !)

Chaque oscillateur est en fait un générateur d'adresses, c'est-à-dire que lorsque le DOC lui donne la parole, il élabore une adresse de 16 bits à l'aide des informations de commandes que l'utilisateur a placées dans ses registres. Vous aurez sûrement deviné que cette adresse désigne un octet de la RAM privative du DOC et que c'est justement cet octet-là qui va ensuite passer dans le convertisseur numérique-analogique qui conduit à la broche 3 du connecteur.

La génération d'adresse

Entrons maintenant dans les détails de la génération d'adresse. Tout d'abord, notons que les calculs d'adresse se font sur 24 bits. À chaque oscillateur est associée une valeur courante sur 24 bits qui est mise à jour à chaque cycle de traitement par addition d'un nombre de 16 bits contenu dans les registres dits de fréquence de cet oscillateur. Les calculs se font dans l'accumulateur de l'unité arithmétique du DOC. Il faut ensuite procéder à un savant mixage pour obtenir l'adresse recherchée en fonction du contenu de l'accumulateur. Cela se fait à l'aide de trois paramètres :

- la résolution ;
- la page de début de la 'table d'onde' ;
- la longueur de la 'table d'onde'.

On peut écrire symboliquement :

```
POUR CHAQUE OSCILLATEUR i ET A
CHAQUE CYCLE DE TRAITEMENT
FAIRE
   ACCUMULATEUR = COURANT(i)
   ACCUMULATEUR = ACCUMULATEUR
                  + INCREMENT(i)
   COURANT(i) = ACCUMULATEUR
   ADRESSE(i) = F(ACCUMULATEUR,
       RESOLUTION(i), DEBUT(i),
       LONGUEUR(i))
FIN TRAITEMENT
```

Parallèlement à l'explicitation de la fonction F, nous développerons un exemple numérique en prenant ACCUMULATEUR = $04FF82. Le paramètre de résolution nous permet déjà de nous débarrasser de 8 bits superflus dans l'accumulateur. On peut garder les bits

de b23 à b8 : $04FF (résolution 7)
de b22 à b7 : $09FF (résolution 6)
etc.
de b16 à b1 : $7FAI (résolution 0)

Vous voyez que dans tous les cas le bit

NUMERI

LOGICIEL DE CALCUL MENTAL

Comprenant 8 niveaux d'additions, 6 de soustractions, 6 de multiplications, 7 de divisions, et des calculs algébriques avec parenthèses : 3000 opérations choisies pour leur intérêt pédagogique, s'adressant aux élèves de CM1, CM2, 6ème et 5ème.

Pour Apple //e (64Ko), Apple //c et Apple IIGS

Prix du logiciel avec sa copie de sauvegarde et sa notice d'emploi : 210 F TTC + 20 F de port

Envoyer ce bon et votre règlement à :
P. Carlier — 16, impasse du bureau — 75011 Paris

0 est sacrifié (le bit 1 aussi d'ailleurs à cause de ce qui suit). Notez tout de même que l'accumulateur est sauvegardé pour le cycle suivant, donc le bit 0 n'est pas du tout inutile !

La 'table d'onde' (wavetable) est tout simplement un segment de la mémoire RAM privative auquel on veut cantonner l'oscillateur. Autrement dit, on ne l'autorise pas à générer des adresses en dehors de ce segment. Pour simplifier les calculs du DOC nous sommes astreints à définir des tables d'onde commençant impérativement en début d'une page (c'est-à-dire avec une adresse divisible par $100) et dont la longueur est un nombre entier de pages, puissance de deux comprise entre 1 et 128 (ce qui correspond à des tailles de 2**8 à 2**15 octets). Notre bonheur ne serait pas complet si on n'ajoutait pas à cela une restriction liant ces deux paramètres, à savoir que le début d'une table d'onde doit avoir une adresse multiple de sa longueur.

Illustration

- un segment d'une page peut commencer à n'importe quel début de page ;
- un segment de 16 pages ne peut commencer qu'à une adresse de la forme $n000 ;
- un segment de 128 pages (alias 32 Ko) ne peut commencer qu'à l'adresse $0000 ou $8000 ;
- un segment de 2**L pages ne peut commencer qu'à une adresse de la forme d*2**(L+8), où d < 2**(8-L).

Plaçons-nous dans le cas d'un segment de 16 pages commençant en $A000 (et finissant en $AFFF par conséquent) et reprenons notre exemple. Avec la résolution la plus fine RES=0, notre adresse $04FF82 est devenue $7FA1. Il nous reste à extraire de ce nombre les 12 bits qui suffiront pour adresser un octet déterminé au sein de la table d'onde. Le DOC choisit les 12 bits de poids fort ($7FA0 -> $7FA) d'où l'adresse définitive $A7FA. De manière plus générale, si on considère un segment de 2**L pages et une résolution RES, l'adresse générée est formée des 8-L premiers bits du numéro de la première page et les 8+L bits suivants sont extraits de l'accumulateur entre les positions 16+RES et 9+RES-L (bornes incluses). Tout ceci est illustré dans le tableau de la page 11 de la documentation développeurs, mais se résume en une seule égalité qui exprime l'adresse générée au k-ième cycle de traitement :

ADRESSE(i) = (DEBUT(i) DIV 2**(8+L))*2**(8+L) + ((k* INCREMENT(i))MOD 2**(8+L)) DIV 2**(9+RES-L))

Cette dernière égalité n'est donnée qu'à titre indicatif. Si elle vous rebute (ce qui ne serait pas étonnant) contentez-vous de comprendre l'explication donnée ci-dessus.

Quelques formules pour mieux saisir

On aura compris que la fréquence du son issu du DOC est une fonction des nombreux paramètres décrits ci-dessus. Commençons par exprimer la fréquence d'échantillonnage (inverse de la période d'échantillonnage) désignée dans la documentation par SR.

```
SR = CLK / (8*(OSC + 2))
```

où l'on retrouve CLK la fréquence d'horloge, 8 le nombre de cycles d'horloge pour un cycle de traitement, OSC le nombre d'oscillateurs autorisés et 2 le nombre de cycles de rafraîchissement. On peut mettre ceci en parallèle avec la fréquence d'échantillonnage d'entrée :

```
SRin = CLK / 8*26
```

où l'on retrouve 26, le nombre de cycles de traitements par numérisation.

Imaginons maintenant que nous définissions une table d'onde en RAM contenant une période unique d'un signal périodique quelconque. Nous pouvons demander à un oscillateur de parcourir cette table indéfiniment en bouclant (croyez-moi sur parole pour le moment). Le son produit sera donc également périodique et de période égale à la durée de parcours de cette table (période du fondamental). Calculons cette durée : à chaque période de traitement, l'oscillateur incrémente l'adresse qu'il produit d'une quantité n :

```
n = FC / (2**(9+RES-L))
```

où FC est l'incrément (contenu des registres dits de fréquence). RES est la résolution et 2**L est la longueur de la table en pages.

La période du fondamental est donc :

T = nombre d'échantillons lus / fréquence de traitement

T = (nombre total / pas de lecture) / SR

T = (2**(L + 8) / n) / SR = 2**(17+RES)/ (FC * SR)

Ce n'est pas un hasard si la longueur de la table n'intervient plus dans cette dernière formule : c'est une conséquence directe des mécanismes compliqués de calcul d'adresse. C'est même vraisemblablement leur cause ! De cette manière, vous avez le choix du pas d'échantillonnage de votre signal : soit vous choisissez un pas très fin et alors vous aurez une table d'onde très longue, soit vous choisissez un pas grossier et alors la même portion du même signal tiendra dans une 'table d'onde' plus courte. Si les paramètres FC, RES et OSC sont identiques dans les deux cas, vous obtiendrez en sortie du DOC un signal de même fréquence mais de qualité sonore peut-être différente.

Les registres dits de commande de fréquence $00-$1F et $20-$3F (frequency control, low and high)

Ceux de la première série contiennent les bits de poids faible et ceux de la seconde les bits de poids fort de l'incrément ajouté à la valeur courante des oscillateurs à chaque cycle de traitement.

Exemple : $05 contient $40 et $25 contient $02 -> incrément de $240 pour l'oscillateur numéro 5

Les registres de volume $40-$5F (volume)

En plus du volume général de la sortie commandé par les 4 bits de poids faible du registre $C03C, on peut commander individuellement et sur une échelle de 0 (mini) à 255 (maxi) le volume de sortie de chaque oscillateur. C'est cela qui permet à la boîte à outils d'offrir des définitions d'enveloppe.

Les registres d'échantillon $60-$7F (data sample)

Chaque registre contient la dernière

valeur lue dans la RAM par l'oscillateur concerné.

Les registres dits d'adresse $80-$9F (adress pointer)

Chaque registre contient le numéro de la première page de la 'table d'onde' à laquelle l'oscillateur est rattaché. Rappelons que la longueur de la table d'onde détermine les bits significatifs de ce registre.

Les registres de commande $A0-$BF (control)

Chaque registre de cette série est composé de plusieurs champs :

b7 à b4 :
canal (ces bits seront présentés sur les broches 4 à 7 du connecteur Molex à chaque fois que l'oscillateur en question sera actif),

b3 :
autorisation de générer une interruption à la fin de chaque cycle de parcours de la table d'onde associée,

b2 à bI :
mode de fonctionnement (voir plus bas),

b0 :
arrêt de l'oscillateur (spontané ou commandé par forçage de ce bit), rappelons que l'oscillateur s'arrête spontanément s'il génère l'adresse d'une mémoire de contenu 0.

Les modes de fonctionnement (commandés par le bit 0) sont 00 (boucle infinie), 01 (une seule itération), 10 (un oscillateur commande le volume ou la durée de la boucle d'un oscillateur de numéro adjacent), 11 (deux oscillateurs de numéros adjacents prennent la main à tour de rôle et indéfiniment).

Les registres de longueur/ résolution $C0-$DF (waveform table size/ resolution/ bank select)

Chaque registre de cette série est composé de plusieurs champs :

b7 à b6 : non utilisés sur IIGS
b5 à b3 : paramètre L (2**L = nombre de pages de la "table d'onde")
b2 à b0 : paramètre RES

Nous ne revenons pas sur leur signification.

Mise en pratique

Nous allons nous contenter de déterminer par le raisonnement les paramètres à utiliser pour que le DOC nous 'rejoue' le contenu de la mémoire que nous venons d'y insérer grâce au programme de numérisation.

Tout d'abord, on remarque qu'un seul oscillateur ne peut pas parcourir les 64Ko d'un coup. On pourrait penser à lui faire parcourir les 32 premiers Ko, lui demander de générer une interruption, traiter cette interruption en modifiant ses registres pour lui faire parcourir le reste de la RAM. C'est faisable mais il existe une technique beaucoup plus naturelle pour le DOC, à savoir l'utilisation du mode relais (swap mode, %11) qui permet d'enchaîner le fonctionnement de deux oscillateurs sans aucune discontinuité. Le premier oscillateur travaillera donc sur les adresses $0000 à $7FFF et le second sur $8000 à $FFFF.

Remarque : ce mode nécessite que le premier oscillateur soit de numéro pair et que le second soit de numéro immédiatement supérieur.

Nous voulons que la période d'échantillonnage en sortie soit identique à celle que nous avons utilisée en entrée. En supposant que l'on a échantillonné l'entrée avec la fréquence maximale SRin, on obtient :

```
OSC = 24
```

Nous voulons également qu'à chaque période d'échantillonnage, l'adresse produite augmente de n=1. Comme L=7, il vient :

```
FC = 2**(2 + RES)
```

Nous pouvons donc choisir ces deux derniers paramètres avec un degré de liberté : RES=0 et FC=$0004 par exemple. Le volume de chaque oscillateur sera réglé à une valeur identique, $FF par exemple.

```
*340 : E1 2E (24 oscillateurs)
  40 FF 41 FF (volume)
  00 04 20 00 01 04 21 00 (fréqu)
  80 00 81 80 (page de début)
  C0 38 C1 38 (longueur/résolution)
  A0 06 A1 07 (commande)
  FF (fin des directives)
*320G (chargement des registres)
*300G 2000.20FF (relire et vérifier)
```

Normalement, dès l'exécution de la routine de chargement des registres vous devriez entendre le contenu sonore de la RAM privative répété sans relâche. Si cela vous convient, vous pouvez continuer à travailler en musique (un peu lancinant tout de même) ou arrêter le DOC, soit par un reset, soit par

```
*340 : A0 01 A1 01 A0 01 A1 01
   FF N 320G
```

La répétition de la séquence semble nécessaire dans certains cas.

Remarque : en dépit des savants calculs ci-dessus, la fréquence restituée par le DOC est trop grande. Ceci est dû au fait que nous avons supposé que la fréquence d'échantillonnage est égale à la fréquence maximum possible, ce qui est approximatif dans le cas d'une horloge à 2,8 MHz et carrément faux dans le cas de l'horloge à 1 MHz. Ajuster la fréquence par l'intennédiaire des paramètres RES ou FC ne peut pas suffire car la modification de l'un de ces paramètres revient toujours à doubler ou diviser par deux la fréquence du son. On peut, en revanche, jouer sur le nombre d'oscillateurs pour régler finement la fréquence. Les paramètres suivants sont satisfaisants :

```
2.8 MHz OSC=27 RES=0 FC=$0004
1 MHz OSC=22   RES=0 FC=$0002
```

Exercice 6.1 : calculez k (ralentissement) en fonction des résultats ci-dessus (résultat : k=10 environ).

Exercice 6.2 : essayez différentes valeurs de RES et de FC satisfaisant à l'égalité FC = 2**(2 + RES).

Exercice 6.3 : utilisez quatre oscillateurs (2 paires) en choisissant leurs numéros assez éloignés pour obtenir un léger décalage entre les deux 'voies' (ex: 0-1 et 13-14). Vous constaterez peut-être un léger écho. Comment produire un décalage plus important ?

Exercice 6.4 : modifier la routine de numérisation pour qu'elle boucle indéfiniment sur elle même (prévoir quand même une sortie par appui sur ⌘) et lancer cette routine tandis que les oscillateurs sont en fonctionnement.

Qu'obtient-on ? Observez les conséquences du léger désajustement des fréquences d'échantillonnage de lecture et de numérisation.

Impression :

Polices 001, 002 & 003

Nouvelles disquettes Domaine Public distribuées par Pom's : des polices de caractères pour la plupart issues du Macintosh, converties pour l'Apple //.

Ces polices de caractères, livrées sur disquettes 5,25' ou 3,5', sont utilisables avec des programmes du type TimeOut SuperFonts, Publish-It !, 816 Paint //e etc.

Les quelques exemples ci-contre ont été réalisés à l'aide de TimeOut SuperFonts sur AppleWorks 1.4. À noter que SuperFonts pratique comme le Macintosh : pour imprimer en 12 points par exemple, il recherche en priorité la police 24 points et s'il la trouve, il la réduit à 12 points gagnant ainsi en finesse. Une fois imprimé, il est impossible de savoir si un Mac ou un Apple // avec SuperFonts a travaillé...

POLICES 01 : 41 polices

Aba Dump 18 — Abbeydecco 14 28 — Aberdeen 12 24 — Akashi 12 — Alexis 24 — Alice 09 12 18 24 — Amslan 18 — Andover 12 — Apl 12 — Apple I 18 — Arabic 24 — Ascii 12 — Asl Finger 14 18 24 36 — Athens 18 — Austin 09 — Austin Econ 09 10 12 18 20 24 — Avant Garde 10 12 14 18 24 — Babylon 09 18 — Barnett 12 18 — Basel 24 48 — Berkeley 18

POLICES 02 : 33 polices

Beta Italic 18 24 36 48 — BlockBuster 36 — Boise 09 18 — Bombay 12 24 — Book 36 — Bookman 10 12 14 18 24 — Border 12 — Boston 07 08 09 10 12 14 16 18 20 24 — Bouton 24 — Brennero 12 24 — Broadway 24 — Broadway E 14 18 — Bubbles 14

POLICES 03 : 35 polices

Bubbles 24 — Cairo 09 18 36 — Calligraphy 24 36 72 — Camelot 12 18 24 — Candy 24 — CapCanaveral 09 — Carmel 12 18 24 — Cartoon 12 — Casey 12 24 — Cavanaugh 18 36 — Celtic 72 — Century 18 24 — Chicago Night 12 — Chicago 12 18 24 — ChicMath 09 12 18 24 — Chivalry 18 — Chugach 12 — Ciao 12 24

Charger les accessoires :

Da.Mover

F. Rozay

N'avez vous jamais eu besoin d'un accessoire de bureau (Classique ou Nouveau) alors que celui-ci ne se trouvait pas sur la disquette de démarrage. Votre disquette système est-elle tellement pleine que vous ne pouvez pas y stocker vos mini-applications (desk accessories) favorites. Un redémarrage de l'Apple IIGS prend tellement de temps que cela vous rebute. Le but de ce programme est de pallier cet inconvénient en permettant de charger des NDA et CDA (entendez par là New Desk Accessories et Classic Desk Accessories) sans que ceux-ci ne soient sur la disquette de démarrage.

'MOVER'
FONCTIONNE
SOUS PRODOS 16
SUR LES Apple IIGS

Utilisation du programme

L'utilisation de ce petit programme est rendue très simple par l'utilisation d'une fenêtre de dialogue ne comportant que quatre boutons.

Les boutons *Lecteur*, *Ouvrir*, *Fermer* ont leur rôle habituel, c'est-à-dire de passer d'un lecteur à un autre, d'ouvrir un répertoire ou de charger un fichier, et de fermer un répertoire. Le bouton *Quitter* permet quant à lui de quitter le programme.

Pour mettre un nouvel accessoire de bureau en mémoire, il suffit, après avoir sélectionné le répertoire contenant les fichiers des DA, de cliquer deux fois sur le nom de celui-ci ou de cliquer une fois pour le sélectionner puis de cliquer dans le bouton Ouvrir. À partir de cet instant l'accessoire est en mémoire et est accessible de l'utilisateur mais n'est pas encore complètement opérationnel. Pour qu'il le devienne il faut quitter le programme.

À propos du programme

Dans le bloc de données nommé ToolTable, la version de chaque outil à charger est indiquée. Celles-ci ne sont valables que pour les personnes qui utilisent la version 3.1 (En attendant la 3.2) du Finder ou les Tools qui sont sur cette disquette. Pour ceux qui font l'erreur de ne pas utiliser cette version, il faut mettre 0 ou un numéro de version inférieur.

Exemple:

```
dc i2'23,$0200'; S.F.O
Version 2.0
```

Si vous voulez utiliser une version antérieure du Standart File Operation, mettre votre numéro de version à la place de $0200, mais ceci à vos risques et périls.

Le programme ne vérifie pas si l'accessoire de bureau demandé est déjà en mémoire. Ceci permet aux éventuels programmeurs d'accessoires de charger différentes versions de leur œuvre.

Icone/Mover est à mettre dans le répertoire Icons de la disquette contenant le Finder. C'est l'icône représentant ce programme.

Source 'MOVER.SRC'

```
        mcopy macs

init    start
        using Data
        using prodos
        using ToolTable

* Bank de donnees =
* bank de programme ==>
* utilisation de l'adressage
* absolu
        phk
        plb

* Sauve l'adresse de
* la page zero
        tdc
        sta     MyZP

* Demarre le Tool Locator
        _TLStartup

* Demarre le Memory Manager
        PushWord £0
        _MMStartup
        PullWord MyID

* Prend de la memoire en Page 0
* pour les Tools
        PushLong £0
        PushLong £$700
        PushWord MyID
        PushWord £$C001
        PushLong £0
        _NewHandle
        pla
        sta     0
        sta     ZPHandle
        pla
        sta     2
```

```
        sta     ZPHandle+2
        lda     °0$
        sta     4

* Quickdraw en route ?
        PushWord £0
        _QDStatus
        pla
        bne     Quickroute

* Non, alors demarre
* Quickdraw II
        lda     4
        pha
        PushWord £$80
        PushWord £0
        PushWord MyID
        _QDStartup

* Oui, alors ne demarre que
* l'Event Manager
Quickroute anop
        lda     4
        clc
        adc     £$300
        pha
        PushWord £255
        PushWord £0
        PushWord £640
        PushWord £0
        PushWord £200
        PushWord MyID
        _EMStartup

* Place le curseur
        PushWord £10
        PushWord £20
        _MoveTo

* Fond noir, ecriture blanche
        PushWord £0
        _SetBackColor
        PushWord £$f
        _SetForeColor

* Affiche une phrase pour
* faire patienter
        PushLong £WaitString
        _DrawString

* Demarre le Desk Manager
        _DeskStartup

* Charge les Tools
        PushLong £ToolTable
        _LoadTools

* Affiche un beau fond bleu
        PushWord £$dddd
        _ClearScreen

* Demarre Window Manager
        PushWord MyID
        _WindStartup

* Demarre Control Manager
        PushWord MyID
        lda     4
        clc
        adc     £$400
        pha
        _CtlStartup

* Demarre Line Edit
        PushWord MyID
        lda     4
        clc
        adc     £$500
        pha
        _LEStartup

* Demarre Quickdraw Auxiliary
        _QDAuxStartup

* Demarre Dialog Manager
        PushWord MyID
        _DialogStartup

* Demarre Standart File
Operation
        PushWord MyID
        lda     4
        clc
        adc     £$600
        pha
        _SFStartup

* Choisit une fleche comme
* curseur et affiche-la
        _InitCursor
        _ShowCursor
        end

* Affiche une fenetre de
* dialogue standart pour
* charger un fichier.
PutDialog start
        using Dialog
        using prodos
        PushWord £100
        PushWord £35
        PushLong £Prompt
        PushLong £0
        PushLong £TypeList
        PushLong £Template
        PushLong £Hook
        PushLong £ReplyRec
        _SFPGetFile
        lda     Good
        bne     Yabon
        end

* Quitte le programme en fermant
* tous les outils qui ont
* ete ouverts
Quit    start
        using Data
        using Prodos
        _SFShutdown
        _DialogShutdown
        _LEShutdown
        _QDAuxShutdown
        _CtlShutdown
        _WindShutdown
        _DeskShutdown
        _EMShutdown
        _QDShutdown
        PushLong ZPHandle
        _DisposeHandle
        PushWord MyID
        _MMShutDown
        _TLShutdown
        _QUIT QuitParams
        end

* Fait patienter l'utilisateur,
* appelle la routine de
* chargement en memoire puis
* appelle la bonne
* routine d'installation.
Yabon   start
        using Data
        using prodos
        using Window
        _WaitCursor
        PushLong £0
        PushLong £ParamList
        _NewWindow
        lda     1,s
        sta     theWindow
        lda     3,s
        sta     theWindow+2
        _SetPort
        jsl     ContDefProc
        jsr     LoadOne
        PushLong theWindow
        _CloseWindow
        _InitCursor
        PushLong DAHandle
        lda     filetype
        cmp     £$B9
        bne     PasCDA
        _InstallCDA

PasCDA  cmp     £$B8
        bne     PasNDA
        _InstallNDA

PasNDA  anop
        brl     PutDialog
        end

* Charge le Desk Accessory
* choisit grace au System Loader
LoadOne start
        using Data
        using prodos
        PushWord £0
        PushWord £0
        PushLong £0
        PushWord £0
        PushWord £$5000
        PushLong £fullname
        PushWord £1
        _InitialLoad
        PullWord NewID
        PullLong DAAddress
        PullWord Direct
        PullWord DSize
        PushLong £0
        PushLong DAAddress
        _FindHandle
        PullLong DAHandle
        rts
        end
```

```
* Routine de gestion des Items.
Hook      start
          using Data
          using prodos
          using Dialog
          phb
          phk
          plb
          PullLong Return
          PullLong ItemHit
          PullLong DialogPort
          PushLong Return
          plb
          rtl
          end

* Ecrit un message dans la
* fenetre.
ContDefProc start
          using Window
          PushWord £10
          PushWord £20
          _MoveTo
          PushLong £Message
          _DrawString
          rtl
          end


* Quelques definitions.

Define    privdata

ButtonItem    equ 10
ScrollBarItem equ 13
StatText      equ 15
UserItem      equ 20
ItemDisable   equ $8000
          end

* Donnees pour le chargement
* des outils


ToolTable privdata
          dc        i'6'
          dc        i'14,$0201'
          dc        i'16,$0202'
          dc        i'18,$0202'
          dc        i'20,$0200'
          dc        i'21,$0200'
          dc        i'23,$0200'

          end

* Donnees generales

Data      privdata

WaitString str      'Un instant,
                     S.V.P.'

MyZP      ds        2
MyID      ds        2
ROMZP     ds        2
NewID     ds        2
Direct    ds        2
DSize     ds        2
```

Comment faire ?

① Vous n'avez pas la disquette Pom's 38 :

- saisissez et sauvegardez la récapitulation 'MOVER' ;
- sous ProDOS 16, depuis le Finder, par exemple, lancez l'application ;
- l'icône sera le losange traditionnel.

② Vous avez la disquette Pom's 38 :

- dans votre dossier 'Icons', mettez l'icône 'Mover' que vous trouvez dans le dossier 'Icone' de la disquette Pom's ;
- maintenant, l'application 'Mover' a son icône en couleurs.

③ Pour essayer :

- Dans le dossier 'Desk.Accs' de la disquette Pom's, vous avez un CDA et un NDA ; chargez-les avec 'Mover' : plus besoin de redémarrer le GS...

```
ZPHandle ds        4
DAHandle ds        4
DAAddress ds       4
          end

* Donnees pour la fenetre
* de dialogue

Dialog    privdata
          using Define

Template dc        i2'0,0,130
                    ,400'
          dc       i2'-1'
          dc       i4'0'
          dc       i4'Item1,
                    Item2, Item3,
                    Item4'
          dc       i4'Item5,
                    Item6, Item7,
                    Item8'
          dc       i4'Item9'
          dc       i4'0'

Item1     dc       i2'1'
          dc       i2'54,265,66,
                    375'
          dc       i2'ButtonItem'
          dc       i4'OuvrirStr'
          dc       i2'0'
          dc       i2'0'
          dc       i4'0'

Item2     dc       i2'2'
          dc       i2'72,265,84,
                    375'
          dc       i2'ButtonItem'
          dc       i4'FermerStr'
          dc       i2'0'
          dc       i2'0'
          dc       i4'0'

Item3     dc       i2'3'
          dc       i2'28,265,40,
                    375'
          dc       i2'ButtonItem'
          dc       i4'LecteurStr'
          dc       i2'0'
          dc       i2'0'
          dc       i4'0'

Item4     dc       i2'4'
          dc       i2'98,265,110,
                    375'
          dc       i2'ButtonItem'
          dc       i4'QuitterStr'
          dc       i2'0'
          dc       i2'0'
          dc       i4'0'

Item5     dc       i2'5'
          dc       i2'28,214,110,
                    239'
          dc       i2'
                    ScrollBarItem'
          dc       i4'0'
          dc       i2'0'
          dc       i2'3'
          dc       i4'0'

Item6     dc       i2'6'
          dc       i2'12,15,24,
                    395'
          dc       i2'UserItem'
          dc       i4'0'
          dc       i2'0'
          dc       i2'0'
          dc       i4'0'

Item7     dc       i2'7'
          dc       i2'28,15,110,
                    215'
          dc       i2'UserItem'
```

```
        dc      i4'0'
        dc      i2'0'
        dc      i2'0'
        dc      i4'0'

Item8   dc      i2'8'
        dc      i2'0,15,12,
                 395'
        dc      i2'UserItem'
        dc      i4'0'
        dc      i2'0'
        dc      i2'0'
        dc      i4'0'

Item9   dc      i2'9'
        dc      i2'115,15,130,
                 395'
        dc      i2'StatText+
                 ItemDisable'
        dc      i4'VersionStr'
        dc      i2'0'
        dc      i2'0'
        dc      i4'0'

TypeList dc     i1'2,$B8,$B9'

OuvrirStr  str  'Ouvrir'
FermerStr  str  'Fermer'
LecteurStr str  'Lecteur'
QuitterStr str  'Quitter'
Prompt     str  'Choisissez un
                 accessoire de
                 bureau'
VersionStr str  'Version
                 VF-1.2
                 (C) 1988,
                 Fred. Rozay'

Return     ds   4
ItemHit    ds   4
DialogPort ds   4

        end

* Donnees pour la
* fenetre d'attente

Window  privdata

ParamList dc    i2'ParamEnd-
                 ParamList'
        dc      i2'%00100000
                 01100000'
        dc      i4'0'
        dc      i4'0'
        dc      i2'0,0,0,0'
        dc      i4'0'
        dc      i2'0,0'
        dc      i2'0,0'
        dc      i2'0,0'
        dc      i2'0,0'
        dc      i2'0,0'
        dc      i4'0'
        dc      i2'0'
        dc      i4'0'
        dc      i4'0'
        dc      i4'
                 ContDefProc'
        dc      i2'84,100,116,
                 500'
        dc      i4'-1'
        dc      i4'0'
ParamEnd anop

Message str     'Veuillez
                 patienter
                 quelques
                 petites
                 secondes...'
theWindow ds    4

        end

* Donnees pour les appels
* a ProDOS/16
prodos  privdata

QuitParams anop
        dc      i6'0'

ReplyRec anop
Good     ds     2
filetype ds     2
auxfltyp ds     2
filename ds     16
fullname ds     128

        end
```

Fichier 'MACS'

```
 MACRO
&lab _CtlStartUp
&lab ldx £$0210
 jsl $E10000
 MEND
 MACRO
&lab _CtlShutDown
&lab ldx £$0310
 jsl $E10000
 MEND
 MACRO
&lab _DeskStartUp
&lab ldx £$0205
 jsl $E10000
 MEND
 MACRO
&lab _DeskShutDown
&lab ldx £$0305
 jsl $E10000
 MEND
 MACRO
&lab _InstallNDA
&lab ldx £$0E05
 jsl $E10000
 MEND
 MACRO
&lab _InstallCDA
&lab ldx £$0F05
 jsl $E10000
 MEND
 MACRO
&lab _DialogStartUp
&lab ldx £$0215
 jsl $E10000
 MEND
 MACRO
&lab _DialogShutDown
&lab ldx £$0315
 jsl $E10000
 MEND
 MACRO
&lab _EMStartUp
&lab ldx £$0206
 jsl $E10000
 MEND
 MACRO
&lab _EMShutDown
&lab ldx £$0306
 jsl $E10000
 MEND
 MACRO
&lab _LEStartUp
&lab ldx £$0214
 jsl $E10000
 MEND
 MACRO
&lab _LEShutDown
&lab ldx £$0314
 jsl $E10000
 MEND
 MACRO
&lab _InitialLoad
&lab ldx £$0911
 jsl $E10000
 MEND
 MACRO
&lab _TLStartUp
&lab ldx £$0201
 jsl $E10000
 MEND
 MACRO
&lab _TLShutDown
&lab ldx £$0301
 jsl $E10000
 MEND
 MACRO
&lab _LoadTools
&lab ldx £$0E01
 jsl $E10000
 MEND
 MACRO
&lab _MMStartUp
&lab ldx £$0202
 jsl $E10000
 MEND
 MACRO
&lab _MMShutDown
&lab ldx £$0302
 jsl $E10000
 MEND
 MACRO
&lab _NewHandle
&lab ldx £$0902
 jsl $E10000
 MEND
 MACRO
&lab _DisposeHandle
&lab ldx £$1002
 jsl $E10000
 MEND
 MACRO
&lab _FindHandle
&lab ldx £$1A02
 jsl $E10000
 MEND
 MACRO
&lab _QUIT &params
```

```
&lab jsl $E100A8
 dc i2"$29"
 dc i4"&params"
 MEND
 MACRO
&lab _QDStartUp
&lab ldx £$0204
 jsl $E10000
 MEND
 MACRO
&lab _QDShutDown
&lab ldx £$0304
 jsl $E10000
 MEND
 MACRO
&lab _QDStatus
&lab ldx £$0604
 jsl $E10000
 MEND
 MACRO
&lab _ClearScreen
&lab ldx £$1504
 jsl $E10000
 MEND
 MACRO
&lab _SetPort
&lab ldx £$1B04
 jsl $E10000
 MEND
 MACRO
&lab _MoveTo
&lab ldx £$3A04
 jsl $E10000
 MEND
 MACRO
&lab _ShowCursor
&lab ldx £$9104
 jsl $E10000
 MEND
 MACRO
&lab _SetForeColor
&lab ldx £$A004
 jsl $E10000
 MEND
 MACRO
&lab _SetBackColor
&lab ldx £$A204
 jsl $E10000
 MEND
 MACRO
&lab _DrawString
&lab ldx £$A504
 jsl $E10000
 MEND
 MACRO
&lab _InitCursor
&lab ldx £$CA04
 jsl $E10000
 MEND
 MACRO
&lab _SFStartUp
&lab ldx £$0217
 jsl $E10000
 MEND
 MACRO
&lab _SFShutDown
&lab ldx £$0317
 jsl $E10000
 MEND
 MACRO
&lab _SFPGetFile
&lab ldx £$0B17
 jsl $E10000
 MEND
 MACRO
&lab _WindStartUp
&lab ldx £$020E
 jsl $E10000
 MEND
 MACRO
&lab _WindShutDown
&lab ldx £$030E
 jsl $E10000
 MEND
 MACRO
&lab _NewWindow
&lab ldx £$090E
 jsl $E10000
 MEND
 MACRO
&lab _CloseWindow
&lab ldx £$0B0E
 jsl $E10000
 MEND
 MACRO
&lab pulllong &addr1,&addr2
&lab ANOP
 AIF C:&addr1=0,.a
 AIF C:&addr2=0,.b
 LCLC &C
&C AMID &addr1,1,1
 AIF "&C"="°",.zeropage
 pullword &addr1
 sta &addr2
 pullword &addr1+2
 sta &addr2+2
 MEXIT
.a
 pullword
 pullword
 MEXIT
.b
 pullword &addr1
 pullword &addr1+2
 MEXIT
.zeropage
 ldy £&addr2
 pullword &addr1,y
 ldy £&addr2+2
 pullword &addr1,y
 MEND
 MACRO
&lab pullword &SYSOPR
&lab ANOP
 pla
 AIF c:&SYSOPR=0,.end
 sta &SYSOPR
.end
 MEND
 MACRO
&lab pushlong &addr,&offset
&lab ANOP
 LCLC &C
 LCLC &REST
&C AMID &addr,1,1
 AIF "&C"="£",.immediate
 AIF "&C"="°",.zeropage
 AIF C:&offset=0,.nooffset
 AIF "&offset"="s",.stack
 pushword &addr+2,&offset
 pushword &addr,&offset
 MEXIT
.nooffset
 pushword &addr+2
 pushword &addr
 MEXIT
.immediate
&REST AMID &addr,2,L:&addr-1
 dc I1'$F4',I2'(&REST)ù-16'
 dc I1'$F4',I2'&REST'
 MEXIT
.stack
 pushword &addr+2,s
 pushword &addr+2,s
 MEXIT
.zeropage
 ldy £&offset+2
 pushword &addr,y
 ldy £&offset
 pushword &addr,y
 MEND
 MACRO
&lab pushword &SYSOPR
&lab ANOP
 AIF c:&SYSOPR=0,.b
 LCLC &C
&C AMID "&SYSOPR",1,1
 AIF ("&C"="£").AND.(S:LONGA),
                        .immediate
 lda &SYSOPR
 pha
 MEXIT
.b
 pha
 MEXIT
.immediate
 LCLC &REST
 LCLA &BL
&BL ASEARCH "&SYSOPR"," ",1
 AIF &BL>0,.a
&BL SETA L:&SYSOPR+1
.a
&REST AMID "&SYSOPR",2,&BL-2
 dc I1'$F4',I2'&REST'
 MEND
 MACRO
&lab str &string
&lab dc i1'1:&string',c'&string'
 MEND
 MACRO
&lab _QDAuxStartup
&lab ldx £$0212
 jsl $E10000
 MEND
 MACRO
&lab _QDAuxShutdown
&lab ldx £$0312
 jsl $E10000
 MEND
 MACRO
&lab _WaitCursor
&lab ldx £$0A12
 jsl $E10000
 MEND
```

Fichier 'BUILD'

```
asml mover.src keep=mover

filetype mover s16
```

Récapitulation 'MOVER'

Après avoir saisie ce code sous moniteur, faites :
create MOVER,t$b3
puis :
bsave MOVER,a$2000,l$800,t$b3

```
2000:04 00 00 00 00 00 00 00 C6 05 00 00 00 0A 04 01
2010:00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00
2020:00 00 01 00 00 00 00 00 2C 00 40 00 20 20 20 20
2030:20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20
2040:F2 26 05 00 00 4B AB 7B 8D 09 03 A2 01 02 22 00
2050:00 E1 F4 00 00 A2 02 02 22 00 00 E1 68 8D 0B 03
2060:F4 00 00 F4 00 00 F4 00 00 F4 00 07 AD 0B 03 48
2070:F4 01 C0 F4 00 00 F4 00 00 A2 02 09 22 00 00 E1
2080:68 85 00 8D 15 03 68 85 02 8D 17 03 A7 00 85 04
2090:F4 00 00 A2 04 06 22 00 00 E1 68 D0 14 A5 04 48
20A0:F4 80 00 F4 00 00 AD 0B 03 48 A2 04 02 22 00 00
20B0:E1 A5 04 18 69 00 03 48 F4 FF 00 F4 00 00 F4 80
20C0:02 F4 00 00 F4 C8 00 AD 0B 03 48 A2 06 02 22 00
20D0:00 E1 F4 0A 00 F4 14 00 A2 04 3A 22 00 00 E1 F4
20E0:00 00 A2 04 A2 22 00 00 E1 F4 0F 00 A2 04 A0 22
20F0:00 00 E1 F4 00 00 F4 F6 02 A2 04 A5 22 00 00 E1
2100:A2 05 02 22 00 00 E1 F4 00 00 F4 DC 02 A2 01 0E
2110:22 00 00 E1 F4 DD DD A2 04 15 22 00 00 E1 AD 0B
2120:03 48 A2 0E 02 22 00 00 E1 AD 0B 03 48 A5 04 18
2130:69 00 04 48 A2 10 02 22 00 00 E1 AD 0B 03 48 A5
2140:04 18 69 00 05 48 A2 14 02 22 00 00 E1 A2 12 02
2150:22 00 00 E1 AD 0B 03 48 A2 15 02 22 00 00 E1 AD
2160:0B 03 48 A5 04 18 69 00 06 48 A2 17 02 22 00 00
2170:E1 A2 04 CA 22 00 00 E1 A2 04 91 22 00 00 E1 F4
2180:64 00 F4 23 00 F4 00 00 F4 50 04 F4 00 00 F4 00
2190:00 F4 00 00 F4 2F 04 F4 00 00 F4 21 03 F4 00 00
21A0:F4 9C 02 F4 00 00 F4 30 05 A2 17 0B 22 00 00 E1
21B0:AD 30 05 D0 6A A2 17 03 22 00 00 E1 A2 15 03 22
21C0:00 00 E1 A2 14 03 22 00 00 E1 A2 12 03 22 00 00
21D0:E1 A2 10 03 22 00 00 E1 A2 0E 03 22 00 00 E1 A2
21E0:05 03 22 00 00 E1 A2 06 03 22 00 00 E1 A2 04 03
21F0:22 00 00 E1 AD 17 03 48 AD 15 03 48 A2 02 10 22
2200:00 00 E1 AD 0B 03 48 A2 02 03 22 00 00 E1 A2 01
2210:03 22 00 00 E1 22 A8 00 E1 29 00 2A 05 00 00 A2
2220:12 0A 22 00 00 E1 F4 00 00 F4 00 00 F4 00 00 F4
2230:A8 04 A2 0E 09 22 00 00 E1 A3 01 8D 26 05 A3 03
2240:8D 28 05 A2 04 1B 22 00 00 E1 22 C1 02 00 20 48
2250:02 AD 28 05 48 AD 26 05 48 A2 0E 0B 22 00 00 E1
2260:A2 04 CA 22 00 00 E1 AD 1B 03 48 AD 19 03 48 AD
2270:32 05 C9 B9 00 D0 07 A2 05 0F 22 00 00 E1 C9 B8
2280:00 D0 07 A2 05 0E 22 00 00 E1 82 F2 FE F4 00 00
2290:F4 00 00 F4 00 00 F4 00 00 F4 00 00 F4 00 50 F4
22A0:00 00 F4 46 05 F4 01 00 A2 11 09 22 00 00 E1 68
22B0:8D 0F 03 68 8D 1D 03 68 8D 1F 03 68 8D 11 03 68
22C0:8D 13 03 F4 00 00 F4 00 00 AD 1F 03 48 AD 1D 03
22D0:48 A2 02 1A 22 00 00 E1 68 8D 19 03 68 8D 1B 03
22E0:60 8B 4B AB 68 8D 9C 04 68 8D 9E 04 68 8D A0 04
22F0:68 8D A2 04 68 8D A4 04 68 8D A6 04 AD 9E 04 48
2300:AD 9C 04 48 AB 6B F4 0A 00 F4 14 00 A2 04 3A 22
2310:00 00 E1 F4 00 00 F4 F6 04 A2 04 A5 22 00 00 E1
2320:6B 06 00 0E 00 01 02 10 00 02 02 12 00 02 02 14
2330:00 00 02 15 00 00 02 17 00 00 02 12 55 6E 20 69
2340:6E 73 74 61 6E 74 2C 20 53 2E 56 2E 50 2E 00 00
2350:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
2360:00 00 00 00 00 00 00 00 00 00 82 00 90 01 FF FF
2370:00 00 00 00 57 03 00 00 6F 03 00 00 87 03 00 00
2380:9F 03 00 00 B7 03 00 00 CF 03 00 00 E7 03 00 00
2390:FF 03 00 00 17 04 00 00 00 00 00 00 01 00 36 00
23A0:09 01 42 00 77 01 0A 00 32 04 00 00 00 00 00 00
23B0:00 00 00 00 02 00 48 00 09 01 54 00 77 01 0A 00
23C0:39 04 00 00 00 00 00 00 00 00 00 00 03 00 1C 00
23D0:09 01 28 00 77 01 0A 00 40 04 00 00 00 00 00 00
23E0:00 00 00 00 04 00 62 00 09 01 6E 00 77 01 0A 00
23F0:48 04 00 00 00 00 00 00 00 00 00 00 05 00 1C 00
2400:D6 00 6E 00 EF 00 0D 00 00 00 00 00 00 00 03 00
2410:00 00 00 00 06 00 0C 00 0F 00 18 00 8B 01 14 00
2420:00 00 00 00 00 00 00 00 00 00 00 00 07 00 1C 00
2430:0F 00 6E 00 D7 00 14 00 00 00 00 00 00 00 00 00
2440:00 00 00 00 08 00 00 00 0F 00 0C 00 8B 01 14 00
2450:00 00 00 00 00 00 00 00 00 00 00 00 09 00 73 00
2460:0F 00 82 00 8B 01 0F 80 73 04 00 00 00 00 00 00
2470:00 00 00 00 02 B8 B9 06 4F 75 76 72 69 72 06 46
2480:65 72 6D 65 72 07 4C 65 63 74 65 75 72 07 51 75
2490:69 74 74 65 72 22 43 68 6F 69 73 69 73 73 65 7A
24A0:20 75 6E 20 61 63 63 65 73 73 6F 69 72 65 20 64
24B0:65 20 62 75 72 65 61 75 28 56 65 72 73 69 6F 6E
24C0:20 56 46 2D 31 2E 32 20 20 20 20 20 28 43 29 20
24D0:31 39 38 38 2C 20 46 72 65 64 2E 20 52 6F 7A 61
24E0:79 00 00 00 00 00 00 00 00 00 00 00 4E 00 60
24F0:20 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
2500:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
2510:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
2520:00 00 00 00 00 00 00 C1 02 00 00 54 00 64 00 74
2530:00 F4 01 FF FF FF FF 00 00 00 00 2F 56 65 75 69
2540:6C 6C 65 7A 20 70 61 74 69 65 6E 74 65 72 20 71
2550:75 65 6C 71 75 65 73 20 70 65 74 69 74 65 73 20
2560:73 65 63 6F 6E 64 65 73 2E 2E 2E F1 A0 00 00 00
2570:F5 02 00 04 00 09 03 F5 02 00 19 00 0B 03 F5 02
2580:00 28 00 0B 03 F5 02 00 3F 00 15 03 F5 02 00 45
2590:00 17 03 F5 02 00 62 00 0B 03 F5 02 00 83 00 0B
25A0:03 F5 02 F0 AF 00 F6 02 F5 02 00 B2 00 F6 02 F5
25B0:02 F0 C3 00 DC 02 F5 02 00 C6 00 DC 02 F5 02 00
25C0:DA 00 0B 03 F5 02 00 E5 00 0B 03 F5 02 00 F7 00
25D0:0B 03 F5 02 00 10 01 0B 03 F5 02 00 1B 01 0B 03
25E0:F5 02 F0 41 01 50 04 F5 02 00 44 01 50 04 F5 02
25F0:F0 4D 01 2F 04 F5 02 00 50 01 2F 04 F5 02 F0 53
2600:01 21 03 F5 02 00 56 01 21 03 F5 02 F0 59 01 9C
2610:02 F5 02 00 5C 01 9C 02 F5 02 F0 5F 01 30 05 F5
2620:02 00 62 01 30 05 F5 02 00 6C 01 30 05 F5 02 00
2630:B0 01 17 03 F5 02 00 B4 01 15 03 F5 02 00 BF 01
2640:0B 03 F5 04 00 D6 01 2A 05 F5 02 F0 E8 01 A8 04
2650:F5 02 00 EB 01 A8 04 F5 02 00 F7 01 26 05 F5 02
2660:00 FC 01 28 05 F5 03 00 06 02 C1 02 F5 02 00 0A
2670:02 48 02 F5 02 00 0D 02 28 05 F5 02 00 11 02 26
2680:05 F5 02 00 23 02 1B 03 F5 02 00 27 02 19 03 F5
2690:02 00 2B 02 32 05 F5 02 F0 5B 02 46 05 F5 02 00
26A0:5E 02 46 05 F5 02 00 6C 02 0F 03 F5 02 00 70 02
26B0:1D 03 F5 02 00 74 02 1F 03 F5 02 00 78 02 11 03
26C0:F5 02 00 7C 02 13 03 F5 02 00 85 02 1F 03 F5 02
26D0:00 89 02 1D 03 F5 02 00 95 02 19 03 F5 02 00 99
26E0:02 1B 03 F5 02 00 A1 02 9C 04 F5 02 00 A5 02 9E
26F0:04 F5 02 00 A9 02 A0 04 F5 02 00 AD 02 A2 04 F5
2700:02 00 B1 02 A4 04 F5 02 00 B5 02 A6 04 F5 02 00
2710:B8 02 9E 04 F5 02 00 BC 02 9C 04 F5 02 F0 CF 02
2720:F6 04 F5 02 00 D2 02 F6 04 F5 04 00 2F 03 57 03
2730:F5 04 00 33 03 6F 03 F5 04 00 37 03 87 03 F5 04
2740:00 3B 03 9F 03 F5 04 00 3F 03 B7 03 F5 04 00 43
2750:03 CF 03 F5 04 00 47 03 E7 03 F5 04 00 4B 03 FF
2760:03 F5 04 00 4F 03 17 04 F5 04 00 63 03 32 04 F5
2770:04 00 7B 03 39 04 F5 04 00 93 03 40 04 F5 04 00
2780:AB 03 48 04 F5 04 00 23 04 73 04 F5 04 00 E2 04
2790:C1 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00
27A0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
27B0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
27C0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
27D0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
27E0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
27F0:00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Liste des fichiers de 'Mover' sur la disquette Pom's 38

```
Mover.src .........Source
Mover .............Application
Build .............Fichier
                   d'assemblage
Macs ..............Fichier de macros
Icone/Mover .......Icône couleurs
Desk.accs/Alarme ..Un CDA exemple
Desk.accs/Horloge .Un NDA exemple
```

Evénement :

DG

Les TimeOut en VF

Oui, c'est un événement capital pour tous les Apple // ; les États-Unis le vivent depuis près d'un an déjà.

Éric Weyland vous en parlait dans son long article du numéro 34 de Pom's, en ajoutant avec regret «...la série TimeOut ne marche, et ne marchera sans doute jamais que sur la version américaine 2.0 d'AppleWorks, et pas sur la version française 1.4. Non que cela soit techniquement impossible : mais il faudrait alors adapter et traduire chacune des applications TimeOut présentes et futures, ce qui représente énormément de travail. L'investissement est sans doute trop grand pour être rentable...».

Ce que ni Apple Computer, ni Claris, ni les Beagle Bros eux-mêmes n'ont pu, ni voulu faire, Pom's et un programmeur français l'ont réalisé. Je tiens à saluer le talent exceptionnel de Daniel Lurot, qui a écrit les patches nécessaires à AppleWorks 1.4, et adapté les logiciels TimeOut. Sans lui, tous les utilisateurs français d'AppleWorks 1.4 seraient encore une fois restés 'sur le bord du chemin'. Sans lui, et sans le dévouement des collaborateurs de Pom's.

Comme vous n'avez peut-être pas tous en mémoire l'article d'Éric Weyland, il n'est pas inutile de rappeler d'abord ce que sont ces logiciels TimeOut.

Et il m'est agréable de pouvoir vous en parler, cette fois-ci, non comme de quelque lointain mirage, mais sachant :

- ✔ qu'ils sont là ;
- ✔ qu'ils fonctionnent ;
- ✔ qu'ils sont entièrement francisés ;
- ✔ qu'une documentation française les accompagne.

TimeOut...

...ou la voie royale pour exalter AppleWorks

Deux touches magiques

- ○ ⌘-Escape : vous êtes dans TimeOut ;
- ○ Escape : vous êtes de retour dans AppleWorks.

...va-et-vient immédiat, écrans familiers.

Il était difficile d'imaginer plus simple. Vous ne quittez pas AppleWorks, c'est le menu TimeOut qui vient à vous. Dans la 'fenêtre' qui s'ouvre au milieu de l'écran, la liste des applications disponibles. Vous sélectionnez l'application. Celle-ci se charge, et fait son travail :

☞ S'agit-il de **SuperFonts** ? Vous pourrez imprimer directement vos fichiers AppleWorks dans un choix extraordinaire de polices Macintosh, en caractères français, en faisant varier la taille et le style (italiques, ombré, relief, inverse), à tout moment, même au milieu d'un mot. Pom's propose déjà 3 disquettes, regroupant plus de 100 polices. Trois qualités d'impression. Importation d'images HR et DHR. Une retouche à votre texte ? Faites Escape, et vous voilà revenu dans AppleWorks, dans le fichier de traitement de texte d'où vous étiez partis. Prêts à imprimer ? ⌘-Escape, vous êtes à nouveau dans SuperFonts.

☞ S'agit-il de **Graph** ? Directement de votre tableur, faites ⌘-Escape. Sélectionnez Graph : c'est à peine si l'on remarque que l'écran a un peu changé ; le titre, en haut ; et en bas, une ligne d'options nouvelles. Ah, mais quelles options ! Elles s'ouvrent les unes après les autres, comme des tiroirs (il suffit de cliquer dessus). Elles vous permettent soit de voir à l'écran, soit d'imprimer : des graphiques en barres, en secteurs, en lignes, en barres empilées, en aires, en XY. Avec légendes ou sans légendes. Titrés, et sous-titrés. Avec ou sans quadrillage. Etc., etc. Voulez-vous retoucher votre tableau ? Escape et vous y êtes. Voulez-vous voir comment a changé votre graphique ? ⌘-Escape et vous êtes de retour dans Graph (les options choisies ont été conservées).

☞ S'agit-il de **SideSpread** ? Vous pouvez imprimer vos grands tableaux à l'horizontale, sans crainte de manquer de place. Vous partez directement du tableur, vous faites ⌘-Escape. Les commandes sont familières : Imprimer Tout ? Lignes ? Colonnes ? Bloc ? Faites votre choix, et allez-y. C'est tout. Pas de manipulations délicates, pas de manœuvres piégées. Et pas de caractères bizarres à la place des é, è, ç et à ! Vous disposez de deux polices francisées, chacune en 8 tailles différentes : de quoi tirer des effets de miniaturisation ou de grossissement surprenants.

☞ S'agit-il d'**UltraMacros** ? Deux applications TimeOut lui sont adjointes : un compilateur (emploi instantané, plus besoin de revenir à la disquette d'installation pour modifier les macros résidentes), et Macros Options : tous les réglages imaginables de la souris et du curseur, et la création immédiate de fichiers "Task" : des fichiers de macros prêts à l'emploi, affectés à une tâche spécifique ; on les démarre d'un menu TimeOut sans quitter AppleWorks, ou même avant de booter AppleWorks au moyen d'un sélecteur genre ProSel (car

ce sont des fichiers SYS). Ces deux applications transférées dans TimeOut ont soulagé l'ancêtre, qui n'est nul autre que SuperMacroWorks, et ont permis (grâce à la place récupérée) d'en faire un véritable langage de programmation. SuperMacroWorks et UltraMacros sont les deux 'chefs d'orchestres' sans lesquels il est désormais difficile d'imaginer un AppleWorks amélioré, mais UltraMacros est beaucoup, beaucoup plus puissant.

Voilà donc les grandes vedettes qui vous attendent. Ce sont incontestablement les 'stars'. Mais il y a aussi les seconds rôles, les auxiliaires toujours présents et indispensables. Il n'est pas nécessaire de les acheter à part, ils sont fournis avec les logiciels principaux.

➪ **Utilitaires** est une application qui permet de 'configurer' Graph, SideSpread, SuperFonts et les autres (entendez par là : choisir les polices de caractères, les imprimantes, et divers autres paramètres utiles). Cette application permet aussi de créer instantanément un menu TimeOut supplémentaire pour une nouvelle application TimeOut.

➪ **Data Converter** crée le pont — qui manquait à AppleWorks — entre données du Tableur et de la Base de données. Grâce à cette application, le presse-papiers transforme les tableaux en base de données, et vice versa, sans aucune manipulation particulière.

➪ **Debug** offre le contrôle des variables d'UltraMacros, et l'affichage des macros.

➪ **Valeurs ASCII** donne, par exemple pour Ctrl-I, 9 en décimal, 9 en hexadécimal, 00001001 en binaire, "Tab", et précise qu'à l'écran c'est un "I" en vidéo inverse.

➪ **File Status** indique l'état du bureau AppleWorks.

➪ **AWP to TXT** transforme les fichiers AppleWorks en fichiers 'Text' (pratique pour changer de machine ou de programme).

*Note : **PathFinder** lui-même n'est pas une application TimeOut : c'est une précieuse modification d'AppleWorks, qui installe directement dans le menu* Autres activités *d'AppleWorks les nouvelles options :* Ajouter un sous-catalogue, Retrancher un sous-catalogue, Renommer un fichier, Verrouiller un fichier *et* Déverrouiller un fichier.

D'autres applications TimeOut viendront prochainement s'ajouter à la liste, au fur et à mesure de leurs francisations, et de la traduction de leurs documentations.

DES EXEMPLES ? PAGES 16 & 30...

Le principe du fonctionnement

TimeOut, qui a donné son nom à l'ensemble de ces applications, est une invention d'Alan Bird. C'est un programme-noyau (un "kernel" comme disent les américains) qui s'installe dans AppleWorks en modifiant son amorçage. Une fois qu'il y est, il n'en bouge plus. Quand vous l'aurez installé une fois, vous n'aurez plus besoin de recommencer.

Ce noyau n'est pas lui-même une application TimeOut. Son rôle est de servir 'd'entremetteur' entre AppleWorks et toutes les applications TimeOut qu'il pourra repérer.

Comment se fait cette entremise ?

- tout d'abord, au démarrage de votre AppleWorks, TimeOut va 'ausculter' le disque où se trouvent les applications TimeOut, et mémoriser les noms de toutes les applications présentes ;
- deuxièmement, à l'appel du menu TimeOut par ⌘-Escape, il affichera la liste ainsi mémorisée, dans laquelle vous ferez votre sélection ;
- troisièmement, aussitôt après la sélection, TimeOut chargera et activera l'application choisie ;
- Quatrièmement, par la touche Escape, TimeOut vous ramènera à AppleWorks.

C'est donc un gestionnaire, qui doit connaître les paramètres d'AppleWorks et ceux des applications, et les préserver à chaque appel pour que vous puissiez les retrouver intacts quand vous passerez de l'un à l'autre.

Quant aux applications, elles partagent toutes la faculté d'être appelées par TimeOut à partir d'AppleWorks. Selon leur degré de complexité, elles feront tourner des programmes puissants, ou se contenteront de tâches relativement subalternes. Bien que théoriquement elles puissent être des programmes indépendants d'AppleWorks, leur intérêt est de pouvoir puiser des données directement dans cet immense réservoir qu'est AppleWorks (fichiers Traitement de texte, Base de données, Tableur), pour accomplir des tâches qu'AppleWorks ne sait pas accomplir lui-même. AppleWorks travaille pour elles, et elles travaillent pour lui (pour notre plus grande satisfaction à nous autres, les utilisateurs).

Cela vous donne, soit dit en passant, une idée de l'extrême complexité interne du fonctionnement de TimeOut : la mémoire de l'Apple étant ce qu'elle est, déjà AppleWorks (qui est un programme énorme) devait fonctionner en 'overlay', c'est-à-dire que n'étaient appelés en mémoire active que les 'segments' dont AppleWorks avait besoin à un moment donné. Sans que vous vous en doutiez, AppleWorks se livre continuellement à un jeu de 'chaises musicales' électronique. Là-dessus arrivent TimeOut et ses applications, qui elles aussi entrent dans la danse. Un exercice de haute voltige s'organise, où nul ne doit se trouver au même moment au même endroit, tout en sachant exactement à quel instant 'tendre la main au trapéziste d'en face'. Mais rassurez-vous, tout cela est, comme ont dit, 'transparent pour l'utilisateur' : l'intégration est parfaitement harmonieuse, et TimeOut se conduit avec des commandes simples et naturelles.

TimeOut : configurations

Les applications TimeOut peuvent être configurées pour fonctionner avec VOTRE matériel.

Vous pourrez choisir 48 imprimantes, parmi lesquelles :

Apple DMP	Apple ImageWriter	Apple ImageWriter II	Apple Scribe
Epson AP 80	Epson FX 80	Epson LQ 800	Epson LQ 1000
Epson LQ 1500	Epson LX 80	Epson MX 70	Epson MX 80
Epson MX100	Epson RX 80	Mannesmann 160	Mannesmann 180
Mannesmann Spirit 80	NEC PC 8023	NEC PC 8025	Oki 82
Oki 83	Oki 84	Oki 92	Oki 93

55 interfaces vous sont proposées, parmi lesquelles :

Série //c	Serial to Parallel //c	IIcs port intégré	Apple Centronics
Communications	Apple Parallel	Apple Série	Apple SSC
Epson APL	Graphicard	Grappler	Grappler +
Grappler Serial	Microbuffer II	Versacard Serial	Versacard Parallel

UltraMacros et SuperMacroWorks sont un cas à part !

Un mot tout de suite pour rassurer les possesseurs de SuperMacroWorks : celui-ci est compatible avec TimeOut et la plupart de ses applications (Graph, SideSpread, Superfonts, etc.). Il ne l'est évidemment pas avec TO.Compiler, TO.Macros Options et TO.Debug, qui sont réservés à UltraMacros.

Les macros de SuperMacroWorks et d'UltraMacros fonctionnent même à l'intérieur de TimeOut et de ses applications ! On peut parfaitement saisir au clavier, par exemple, d'une part la série de commandes qui mène d'un fichier tableur dans AppleWorks jusqu'à son affichage sous une forme graphique donnée dans Graph, et d'autre part la série de commandes qui ramène au point de départ. En deux touches seulement on pourra ainsi faire le va-et-vient entre un tableau et sa représentation graphique ! On croit rêver...

Mais alors, pourquoi UltraMacros ? Parce que c'est autre chose. UltraMacros est un vrai langage de programmation, dans lequel on a inclus SMW par souci de continuité. C'est, en quelque sorte, une 'fleur' faite aux habitués de SMW, qui ne seront quasiment pas dérangés dans leurs habitudes. En gros, on peut dire que la souris fonctionne de la même manière (sauf qu'elle est vraiment réglable, en déplacement et en 'clic'), et que la saisie des macros au clavier est pratiquement la même. Mais là ou SMW avait une fonction conditionnelle rudimentaire liée au seul caractère sous le curseur, et une unique variable numérique non directement accessible, UltraMacros a une authentique fonction if-then-else appliquable à 26 variables numériques et 10 chaînes-texte définissables de façon indépendante et variée. D'ailleurs, une gamme entière de nouvelles instructions est conçue pour fonctionner avec ces variables comme paramètres. Ajoutez-y les instructions Peek, Poke et même Call pour les programmeurs les plus audacieux, et vous aurez une petite idée de ce qu'UltraMacros est capable d'accomplir !

Une vue d'ensemble

Le trait génial d'Alan Bird (créateur de TimeOut) est d'avoir conçu, dès le départ, un ensemble à la fois homogène et ouvert.

• Homogène, puisqu'il repose sur un gestionnaire unique, le noyau TimeOut. La règle du jeu a ainsi été fixée une fois pour toutes.

• Ouvert, parce qu'il n'y a pas de limite au nombre des applications que TimeOut pourra appeler, à condition qu'elles respectent cette règle du jeu.

Mais, me direz-vous, cela fait beaucoup de monde à loger dans mon Apple. Comment vais-je y arriver ?

Rassurez-vous, il n'y aura pas de bousculade, tout ce monde n'entrera dans votre Apple que dans la limite de la mémoire disponible.

Avez-vous eu la prévoyance (comme vous l'a recommandé maintes fois Jean-Yves Bourdin) de renforcer la capacité-mémoire de votre Apple ? Parfait. Les applications TimeOut pourront y séjourner à plusieurs, et être appelées quasi-instantanément. Sinon ? Elles patienteront dans la salle d'attente (le "disque d'applications TimeOut"), et le noyau TimeOut les appellera et les fera passer une à une.

Les privilégiés seront évidemment les possesseurs de GS : la mémoire de celui-ci a été dimensionnée en fonction d'applications graphiques tellement voraces par rapport à l'appétit de TimeOut que vous en aurez largement assez pour loger entièrement en RAM, par exemple : AppleWorks tout entier, UltraMacros, SuperFonts et la totalité de ses polices de caractères, plus

quelques autres applications. Il vous restera quand-même un Bureau de plusieurs centaines de Ko.

Mais, même dans la pire des hypothèses, les applications TimeOut fonctionneront quand-même : là ou AppleWorks va, TimeOut va aussi. Au prix, néanmoins, de fréquents appels aux lecteurs de disques. Vous voilà prévenus. La gestion des disques a été améliorée dans une récente version TimeOut (celle, justement, que diffuse Pom's) : une nouvelle option de TO.Utilitaires permet de créer à tout moment, sans quitter AppleWorks, un menu TimeOut supplémentaire, lié à un disque d'applications donné : décidez-vous, par exemple, en cours de travail d'utiliser Graph alors que vous aviez chargé SuperFonts, et que vous disposez de peu de mémoire ? En quelques secondes, par appel à des menus clairs, SuperFonts est reconduit, et Graph le remplace.

Le 'miracle' des TimeOut en français

De ce qui précède vous aurez déjà compris que TimeOut, ses applications, UltraMacros et AppleWorks vivent en étroite symbiose. TimeOut était conçu au départ pour la version américaine 2.0 et la gérait compte tenu de toutes ses adresses internes spécifiques. Dans la version française 1.4 d'AppleWorks, toutes ces adresses sont décalées ou modifiées. Pour reprendre ma métaphore de l'exercice de haute voltige, imaginez un trapéziste qui s'élancerait vers son partenaire sans avoir été prévenu qu'on a rallongé à son insu les courroies de son trapèze... C'est exactement ce qui advient si l'on essaie de faire tourner un TimeOut américain sur un AppleWorks français.

Il fallait donc :

- modifier AppleWorks ;
- modifier TimeOut ;
- modifier les applications TimeOut ;
- et réconcilier, dans le nouvel ensemble obtenu, toutes les adresses internes. Je ne parle pas de la francisation, qui vient s'y ajouter par-dessus le marché.

Tâche impossible ? Quasiment, à moins d'un miracle... C'est justement ce qui s'est produit par un heureux concours de circonstances : Daniel Lurot, le programmeur qui a adapté TimeOut, n'avait nullement ce projet en tête au départ. Il ignorait même tout de TimeOut, et avait commandé aux Beagle Bros un exemplaire de SuperMacroWorks. La destin a voulu qu'on lui envoie par erreur une exemplaire d'UltraMacros, évidemment incompatible avec AppleWorks 1.4.

Or, il se trouve que Daniel avait déjà, pour satisfaire sa curiosité, entièrement désassemblé AppleWorks 1.4. Oui, oui, vous avez bien lu (ça représente deux classeurs pleins de code, d'une épaisseur de quelques 15 centimètres). Il en connaissait donc déjà tous les tours et détours ; adapter UltraMacros ne lui a nullement paru être une tâche surhumaine, et il a même essayé diverses façons de s'y prendre. Puis il en a modestement informé l'auteur Randy Brandt, lequel lui a aussitôt recommandé de me contacter. Ce qu'il fit.

À partir de là, ce qui avait été un heureux coup du sort individuel s'est transformé en travail d'équipe, et un projet plus ambitieux prit corps : pourquoi ne pas adapter, franciser et diffuser, à l'intention de tous les utilisateurs d'AppleWorks 1.4, les plus intéressantes des applications TimeOut ? Pari tenu...

Dans les revues américaines consacrées aux Apple //, on voit en double page deux publicités très éloquentes :

- la première est de Claris (distributeur d'AppleWorks). Elle montre une rutilante Porsche rouge... sur quatre parpaings en guise de roues. Le texte dit «Il y a encore certains possesseurs d'Apple qui n'emploient pas AppleWorks» ;
- la deuxième est des Beagle Bros. On y voit quatre magnifiques roues à jantes de course... sans rien par-dessus. Le texte dit «Il y a encore des utilisateurs d'AppleWorks qui ne connaissent pas TimeOut».

La revue Pom's n'est ni Claris, ni les Beagle Bros, mais grâce à elle, et à l'esprit Apple de son équipe, le bolide complet — et francisé — est à votre disposition. Installez-vous au volant : croyez-moi, l'expérience en vaut la peine ! Je vous souhaite bonne route, et beaucoup de plaisir.

---

*Pom's vous proposera bientôt — en exclusivité également — **FileMaster** (gestion complète de fichiers sous AppleWorks), **DeskTools** (une dizaine d'accessoires de bureau, y compris un cryptage de fichiers), **MacroTools** (de puissantes macros pour AppleWorks).*

Pour les prochains numéros, Dimitri Geystor vous prépare des utilitaires dont il a le secret, sous UltraMacros...

Tarif des logiciels en français — avec manuels français — pour AppleWorks 1.4

Graph : 850 F TTC (carte Joker : 765 F)

SuperFonts (comprenant une quarantaine de polices de caractères) : 800 F TTC (carte Joker : 720 F)

UltraMacros : 650 F TTC (carte Joker : 585 F)

SideSpread : 550 F TTC (carte Joker : 495 F)

PathFinder : 150 FTTC (les 4 TimeOut inclus sont utilisables si vous avez déjà un TimeOut, UltraMacros par exemple)

Le calcul des Premiers :

Prime.Factor

Pascal Cantot

Source du programme 'PRIME FACTOR'

Assembleur Merlin-Pro

```
 1 ********************************
 2 *                              *
 3 *   P R I M E    F A C T O R   *
 4 *   ------------------------   *
 5 *   Par P. CANTOT, 20/05/88    *
 6 *                              *
 7 ********************************
 8
 9
10 N1       =     $0          ;Nombre de 32 bits à tester
11 ADRTBL   =     $4          ;Index vers PREMTBL
12 N2       =     $6          ;Diviseur trouvé (16 bits)
13 R        =     $8          ;Reste (0 si non premier) (16 bits)
14 USRJMP   =     $A          ;Vecteur pour fonction USR()
15 SQRN1    =     $50         ;Racine (par excès) de N1 (16 bits)
16 ARG      =     $A5         ;Registre flottant secondaire
17 RESULT   =     $EB         ;Résultat division (32 bits)
18 TEMP4    =     $FC         ;Temporaire (32 bits)
19
20 PRIME    =     $2000       ;ORG de PRIME FACTOR
21 PREMTBL  =     PRIME+$100  ;Adresse table nbres premiers
22
23 ERROR    =     $D412       ;Affiche erreur X
24 GIVAYF   =     $E2F2       ;A,Y -> FAC
25 GETADR   =     $E752       ;INT(FAC) -> $50
26 MAF      =     $EB66       ;FAC -> ARG
27 SQR      =     $EE8D       ;SQR(FAC) -> FAC
28
29 ********************************
30
31          EXP   OFF
32
33 MOV      MAC               ;MOVe byte
34          LDA   $1
35          STA   $2
36          <<<
37
38 MOV4     MAC               ;MOVe 4-byte word
39          MOV   $1;$2
40          MOV   $1+1;$2+1
41          MOV   $1+2;$2+2
42          MOV   $1+3;$2+3
43          <<<
44
45 SUBD     MAC               ;SUBstract Double-byte (16 bits)
46          LDA   $1
47          SEC
48          SBC   $2
49          STA   $3
50          LDA   $1+1
51          SBC   $2+1
52          STA   $3+1
```

PRIME FACTOR est un petit programme machine qui peut, en quelques secondes — au pire — déterminer les diviseurs premiers d'un nombre compris entre 2 et 1 073 741 824 (2^{30}, soit environ un milliard).

Les applications sont nombreuses :

- déterminer si un nombre est premier ou non ;
- décomposition d'un nombre en produit de facteurs premiers (l'objet de DECOMPOSER), afin de simplifier une fraction, de déterminer le Plus Grand Commun Diviseur de deux nombres, etc. ;
- calcul très simple et très **rapide** de tables de nombres premiers : j'ai ainsi mis sur une disquette tous les nombres premiers jusqu'à 580 000, soit 47 600 nombres. Le calcul a pris 8 heures. En Basic, un calcul identique m'avait pris 48 heures ! Autre exemple : il met 6 secondes seulement pour trouver que 42 841 001 est premier.

De plus, le programme en lui-même présente un intérêt pédagogique, puisqu'il comprend une routine de division entière 32 bits par 16 bits, ainsi qu'une routine de conversion d'un nombre flottant en entier 32 bits...

Mode d'emploi

PRIME.FACTOR s'utilise simplement à partir de l'Applesoft par la commande Y = USR (X) (ou PRINT USR(X)), où X est le nombre entier à tester (il doit être compris entre 2 et 1E9). Y contient alors le **plus petit diviseur premier** de X, ou zéro si X est premier. Notons que PRIME.FACTOR fonctionnera pour

X compris entre 1E9 et 4E9 (plus exactement : 2^{32}), mais seulement si X admet un diviseur inférieur à 32 768. Dans le cas contraire, un "OVERFLOW ERROR" est envoyé.

Exemples :

USR(12) donne 3
USR(11) donne 0 car 11 est premier
USR(20989) donne 139 car 20989 = 139*151
USR(42841001) donne 0 car X est premier
USR(3343549127) donne 31573 car X = 31573*105899
USR(2840783401) donne OVERFLOW ERROR, car X>1E9 et Y = 53299^2

Principe

PRIME.FACTOR utilise pour ses calculs une table de tous les nombres premiers compris entre 2 et 32 768 (PREMTBL), soit 3 512 entiers codés sur 2 octets. Cette table DOIT être chargée 256 octets après le début de PRIME.FACTOR (c'est-à-dire $2100 si PRIME.FACTOR est assemblé en $2000) avant toute utilisation de PRIME.FACTOR.

PRIME.FACTOR se lance depuis le disque par BRUN ou — (prononcer "smart RUN"). Il revectorise alors la commande "USR" du Basic et rend la main au programme appelant.

Lors de chaque appel Y = USR(X), l'argument X est placé dans un accumulateur flottant, FAC. PRIME.FACTOR le convertit alors en nombre entier sur 4 octets, puis le divise successivement par tous les nombres de sa table compris entre 2 et SQR(X), jusqu'à ce que le reste de la division soit nul. Auquel cas, le diviseur est retourné dans FAC, et l'Applesoft se charge ensuite d'affecter le contenu de FAC à Y. Si aucun diviseur n'est trouvé, alors c'est 0 qui est placé dans FAC, puis dans Y. Enfin, si PRIME.FACTOR arrive à la fin de sa table (c'est-à-dire si X > 32768^2, soit environ 1E9), alors un message d'erreur est envoyé.

Note : j'ai privilégié ici la vitesse. Si toutefois 7Ko vous semblent trop

```
53          <<<
54
55 CMPD     MAC                 ;CoMPare Double-byte (16 bits)
56          LDA   $1
57          CMP   $2
58          LDA   $1+1
59          SBC   $2+1          ;C = 1 si $1 >= $2
60          <<<
61
62 ROLD     MAC                 ;ROtate Left Double-byte
63          ROL   $1
64          ROL   $1+1
65          <<<
66
67 ASL4     MAC                 ;Shift 4-byte word
68          ASL   $1
69          ROL   $1+1
70          ROLD  $1+2
71          <<<
72
73 ROL4     MAC                 ;ROtate Left 4-byte word
74          ROLD  $1
75          ROLD  $1+2
76          <<<
77
78
79 ********************************
80
81          ORG   PRIME
82
83 SETUSR   MOV   £>INTFAC32;USRJMP+2 ;Revectorise USR
84          MOV   £<INTFAC32;USRJMP+1
85          RTS
86
87 INTFAC32 JSR   MAF           ;FAC -> ARG
88          LDA   ARG           ;Exposant -> A
89          AND   £$7F
90          STA   CNTR          ;Nbre de bits dans la mantisse
91          MOV   £0;N1         ;Mise à zéro de N1
92          STA   N1+1
93          STA   N1+2
94          STA   N1+3
95
96 INTFAC2  DEC   CNTR          ;Convertit ARG
97          BMI   INTFAC3       ; en un nombre entier N1
98          ASL   ARG+4         ; de 32 bits (4 octets)
99          ROL   ARG+3
100         ROL   ARG+2
101         ROL   ARG+1
102         ROL4  N1
103         JMP   INTFAC2
104
105 INTFAC3 JSR   SQR           ;SQR(FAC) -> FAC
106         JSR   GETADR        ;INT(FAC) -> $50
107         INC   SQRN1         ;S'assure que la valeur
108         BNE   TESTNUM       ; est par excès
109         INC   SQRN1+1
110
111 TESTNUM MOV   £>PREMTBL;ADRTBL+1
112         MOV   £<PREMTBL;ADRTBL
113         LDY   £0
114         BEQ   TEST3         ;Toujours pris
115
116 TESTLUP INY
117         BNE   TEST3
118         INC   ADRTBL+1
119
120 TEST3   LDA   (ADRTBL),Y ;Met le nombre premier suivant
121         STA   N2            ; dans N2
122         BEQ   OVRFLOW       ;Si 00, fin table
```

```
123          INY
124          LDA   (ADRTBL),Y
125          STA   N2+1
126          CMPD  N2;SQRN1     ;N2 > racine(N1)?
127          BCS   FNDPREM      ;Si oui, alors N1 premier!
128
129          JSR   DIV32        ;Effectue N1/N2
130
131          LDA   R            ;Reste nul?
132          BNE   TESTLUP      ;Non -> on continue
133          LDA   R+1
134          BNE   TESTLUP
135
136          LDA   N2+1         ;Si non premier,
137          LDY   N2           ; retourne le diviseur
138 EXIT     JMP   GIVAYF
139
140 FNDPREM  LDA   £0           ;Sinon, retourne 0
141          TAY
142          BEQ   EXIT         ;Toujours pris
143
144 OVRFLOW  LDX   £69          ;Affiche OVERFLOW ERROR
145          JMP   ERROR
146
147 ********************************
148
149 * Division entière de N1 (32 bits) par N2 (16 bits)
150
151 DIV32    MOV4  N1;TEMP4     ;Dividende -> Temp4
152          MOV   £0;R         ;Mise à zéro du reste R
153          STA   R+1
154          LDX   £32          ;Le dividende a 32 bits
155
156 DIVLUP   DEX
157          BMI   DIVEND
158          ASL4  TEMP4
159          ROLD  R
160          CMPD  R;N2         ;Si R>=N2, le quotient est 1
161          BCS   DIV2         ;sinon 0
162          ROL4  RESULT
163          JMP   DIVLUP       ;Toujours pris
164 DIV2     ROL4  RESULT
165          SUBD  R;N2;R       ;R - N2 -> R
166          JMP   DIVLUP
167
168 DIVEND   RTS
169
170 CNTR     DS    1
171          LST   OFF
```

long, vous pouvez toujours modifier PRIME FACTOR en réduisant la table, voire même en la supprimant totalement (on teste alors tous les entiers impairs). Dans ce dernier cas, la mémoire occupée sera très faible (moins de 256 octets), mais alors les temps de calcul deviennent très importants.

Comment créer PREMTBL

Il est, bien sûr, possible de saisir PREMTBL manuellement, à partir du moniteur. Mais alors "bonjour le travail" : PREMTBL est longue de 3512 x 2 = 7024 octets. Le mieux est de calculer PREMTBL à l'aide du programme Basic CRE.PREMTBL. Celui-ci calcule les 3 512 nombres premiers inférieurs contenus dans PREMTBL, puis les code sur 2 octets et enfin les sauve sous forme binaire dans le fichier PREMTBL, en utilisant le préfixe courant.

Pour lancer le calcul, il suffit de taper "RUN CRE.PREMTBL". Attention, cela prend plus d'une heure !

À vous maintenant de construire vos applications, comme par exemple la simplification de fractions ou la représentation graphique de spirales de nombres premiers.

PRIME FACTOR
FONCTIONNE
SUR LES
Apple II+, //e, //c ET IIGS
SOUS DOS 3.3 OU
SOUS PRODOS

Programme 'CRE.PREMTBL'

```
10  REM  CALCUL ET CREATION DE
     PREMTBL
20  LOMEM: 16384
30 D$ =  CHR$ (4): DIM K(3512)
40  PRINT : PRINT "CALCUL PREL
    IMINAIRE";
50 K(2) = 1:K(1) = 0:K(3) = 1:
    N = 2
60  FOR I = 5 TO 255 STEP 2
70  IF  INT (I / 2) = (I / 2)
    THEN K(I) = 0: GOTO 120
80 C =  INT ( SQR (I))
90  FOR J = 3 TO C STEP 2: IF
     INT (I / J) = (I / J) THE
    N J = 999
100  NEXT
110  IF J < 999 THEN  PRINT ".
    ";:K(I) = 1:N = N + 1
120  NEXT
130  PRINT : PRINT : PRINT "CA
    LCUL DE PREMTBL:": PRINT
140 J = 1: FOR I = 1 TO 256: I
    F K(I) THEN K(J) = I:J = J
     + 1: PRINT I,
150  NEXT
160  FOR I = 255 TO 32767 STEP
     2
170  FOR J = 2 TO 54
180  IF  INT (I / K(J)) = (I /
     K(J)) THEN J = 99
190  NEXT
200  IF J < 99 THEN  PRINT I,:
    N = N + 1:K(N) = I
210  NEXT
220  PRINT : PRINT  CHR$ (7);"
    CREATION DE PREMTBL"
230 AD% = 8448: REM $2100
240  FOR I = 0 TO 3511:X = K(I
     + 1)
```

```
250  POKE AD% + I * 2 + 1,X /
     256: POKE AD% + I * 2,X -
     256 *  INT (X / 256): NEXT

260  PRINT  CHR$ (7)
270  PRINT D$"BSAVEPREMTBL,A"A
     D%",L"2 * N
280  TEXT
```

Récapitulation 'PRIME.FACTOR'

Après avoir saisi ce code sous moniteur, vous le sauvegarderez par BSAVE PRIMEFACTOR,A$2000,L$D9

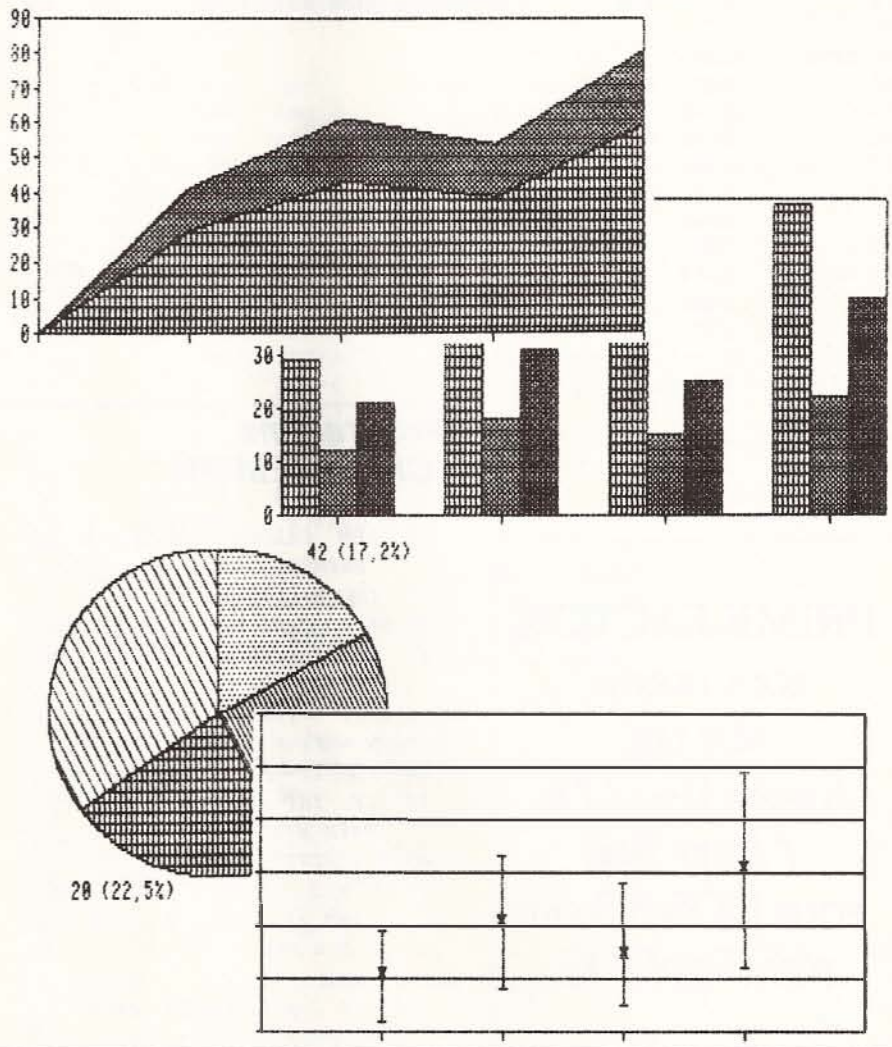
```
2000:A9 20 85 0C A9 09 85 0B
2008:60 20 66 EB A5 A5 29 7F
2010:8D D8 20 A9 00 85 00 85
2018:01 85 02 85 03 CE D8 20
2020:30 13 06 A9 26 A8 26 A7
2028:26 A6 26 00 26 01 26 02
2030:26 03 4C 1D 20 20 8D EE
2038:20 52 E7 E6 50 D0 02 E6
2040:51 A9 21 85 05 A9 00 85
2048:04 A0 00 F0 05 C8 D0 02
2050:E6 05 B1 04 85 06 F0 26
2058:C8 B1 04 85 07 A5 06 C5
2060:50 A5 07 E5 51 B0 12 20
2068:83 20 A5 08 D0 DF A5 09
2070:D0 DB A5 07 A4 06 4C F2
2078:E2 A9 00 A8 F0 F8 A2 45
2080:4C 12 D4 A5 00 85 FC A5
2088:01 85 FD A5 02 85 FE A5
2090:03 85 FF A9 00 85 08 85
2098:09 A2 20 CA 30 39 06 FC
20A0:26 FD 26 FE 26 FF 26 08
20A8:26 09 A5 08 C5 06 A5 09
20B0:E5 07 B0 0B 26 EB 26 EC
20B8:26 ED 26 EE 4C 9B 20 26
20C0:EB 26 EC 26 ED 26 EE A5
20C8:08 38 E5 06 85 08 A5 09
20D0:E5 07 85 09 4C 9B 20 60
20D8:00
```

Programme 'DECOMPOSEUR'

```
1  REM   DECOMPOSITION EN PROD
     UIT DE FACTEURS PREMIERS
10  TEXT : HOME : PRINT "Je ch
     arge PRIME FACTOR et PREMT
     BL..."
20  PRINT  CHR$ (4)"BRUN PRIME
     .FACTOR,A$2000"
30  PRINT  CHR$ (4)"BLOAD PREM
     TBL,A$2100"
40  LOMEM: 16384
50  HOME : PRINT "En utilisant
     PRIME FACTOR, je peux dec
     omposer un nombre entier <
     à 1E9 en": PRINT "produit
     de facteurs premiers."
60  PRINT :MAX = 32768 ^ 2
70  PRINT : PRINT : INPUT "Nom
     bre à décomposer ou 0 ? ";
     X:X =  INT ( ABS (X))
80  PRINT : IF X = 0 THEN  END

90  IF X >  = MAX THEN  PRINT
     "Trop grand!": GOTO 70
100  PRINT "Liste des facteurs
     premiers de "X":",
110 Y =  USR (X): IF Y = 0 THE
     N  PRINT X: PRINT : GOTO 7
     0
120  PRINT Y,:X = X / Y: GOTO
     110
```

TimeOut Graph

Voici quelques exemples qui illustrent la puissance de Graph. Sans quitter AppleWorks, les graphiques se construisent instantanement exactement adaptes a vos besoins. Changez une valeur et trois touches suffisent pour afficher le graphique modifie. Et pourquoi ne pas automatiser construction et impression des graphes avec UltraMacros ?



Applesoft & AppleWorks

Dimitri Geystor

Vous l'avez sûrement entendu dire, avec un brin d'envie, par des amis qui programment : «Un éditeur de programmes Basic ? J'emploie mon traitement de texte, voyons !». Mais comment font-ils donc ?

Le principe est fort simple : partant d'un fichier de traitement de texte, on le transforme en un fichier de type TXT (ASCII), on va en Basic Applesoft, on EXECute ce fichier : à partir de là, on peut le sauver sous le nom qu'on veut avec SAVE : il est devenu un fichier BAS (faites "RUN", vous verrez).

La réalité est un peu plus complexe, surtout si l'envie vous prend d'aller travailler avec AppleWorks un fichier Basic déjà existant. Essayez de transformer directement un fichier BAS en fichier traitement de texte : AppleWorks est d'accord pour convertir n'importe quel fichier en traitement de texte AWP, mais les résultats sont pour le moins déconcertants. Il vous faut d'abord 'traduire' votre fichier type BAS en fichier type TXT (ASCII), et malheureusement il n'existe pas en Basic Applesoft de commande qui fasse le contraire d'EXEC. Il faut donc ruser une peu. On verra cela un peu plus loin.

Tout ceci peut évidemment se maîtriser, et un programme sous forme de traitement de texte est quand même plus facile à travailler que sous sa forme Basic (la procédure de modification en Applesoft avec ESC et les flèches est parfaitement odieuse). Mais, sauf pour les tout petits programmes, ce n'est pas la joie : pour peu qu'il y ait beaucoup d'instructions par numéro de ligne, et des REMs à droite et à gauche, la lecture ne sera pas facile. Non, non, croyez-moi, ceux qui vous disent que le traitement de texte AppleWorks est l'éditeur idéal pour les programmes en Basic Applesoft vous racontent des histoires. Ils 'friment'. Les avez-vous seulement vus à l'œuvre ?

Mais heureusement SuperMacroWorks peut venir mettre de l'ordre dans ce fouillis, en permettant de **formater** et **déformater** votre programme. Ce sera le sujet de cet article, qui vous proposera aussi un certain nombre de macros-commandes pour accélérer et simplifier l'écriture de vos programmes Applesoft. Certes, cela ne vaudra jamais un véritable éditeur Basic écrit exprès en langage machine (comme Program Writer d'Alan Bird, ou le GPLE des Beagle Bros, ou encore E.P.E. de Pom's), mais les avantages sont certains par rapport aux misérables moyens d'édition de l'Applesoft abandonné à lui-même...

En fait, je vous propose deux macros de formatage :

- l'une 'aère' simplement le programme, en allant à la ligne entre chaque instruction, et en intercalant à chaque fois un trait fait de 5 'tirets'. Ce sera la macro <sa-Ctrl-A> du fichier MACROS.BASIC.
- l'autre 'ventile' le programme, en envoyant chaque instruction séparée à la ligne. C'est la macro <sa-Ctrl-V> du fichier MACROS.BASIC.

Il est possible d'aérer et de ventiler à la fois, si vous ne craignez pas les courants d'air. À exécuter impérativement dans l'ordre <sa-Ctrl-A> puis <sa-Ctrl-V>, sinon vous auriez vraiment trop de tirets...

Vous pourrez ensuite vous en donner à cœur-joie :

- imprimer des listings, complets ou partiels ;
- utiliser les fonctions de recherche et de remplacement ;
- ajouter des 'modules' pré-construits, pour vos sous-routines.

Un conseil : il est hautement recommandé d'installer AppleWorks en RAM, car vous devrez le quitter, puis y revenir assez souvent pour tester votre programme. Avec un bon sélecteur (ProSel, par exemple), c'est l'affaire de 2 à 3 secondes. Sinon, l'attente devient vite fastidieuse. Pour automatiser encore plus les choses, compilez MACROS.BASIC avec SMW (les macros seront actives au démarrage d'AppleWorks) puis renommez SUPER.SYSTEM en SUPER.BASIC et faites-le démarrer à la demande avec le sélecteur.

Mais parlons d'abord de la conversion proprement dite, obligatoire et incontournable.

Conversion AWP ➡ Basic, et vice versa

Cette conversion — dans un sens comme dans l'autre — est la condition nécessaire pour pouvoir employer le traitement de texte AppleWorks en tant qu'éditeur de programmes en Basic.

Les étapes sont les suivantes :

✗ Dans le sens AppleWorks ➡ Basic

① Le programme est écrit avec le traitement de texte AppleWorks.

② Le fichier AWP ainsi obtenu est 'imprimé sur disque (ASCII)'. Il devient donc un fichier de type TXT. Supposons qu'il a été sauvé sous le nom PROG.TXT.

③ On quitte AppleWorks, on va en Basic Applesoft.

④ On vérifie que PROG.TXT est bien sur le disque, et on fait EXEC PROG.TXT. Le fichier est chargé en mémoire sous forme d'un programme Basic.

Note : Ne vous inquiétez pas des '§' qui s'impriment à l'écran, c'est normal ; vous verrez peut-être s'imprimer d'autres choses inattendues, et même des messages d'erreur, si votre programme contient par exemple des lignes non numérotées (Voir encadré pour une explication de ce phénomène).

⑤ On peut d'ores et déjà utiliser le programme en faisant "RUN".

⑥ Pour le conserver sous la forme BAS, faire SAVE PROG (où PROG sera le nom définitif du programme en Basic).

✗ Dans le sens Basic ➡ AppleWorks

Il faut commencer par préparer un petit programme 'chargeur', dans AppleWorks, et le sauver sur disque

Macro-commandes 'MACROS.BASIC'

```
AppleWorks 1.4                                  SuperMacroWorks

                          MACROS.BASIC
                          ============

     Cet ensemble de macros permet d'utiliser votre Traitement de
     Textes AppleWorks comme éditeur de programmes Basic Applesoft.

     Pour en tirer le meilleur parti, installez votre AppleWorks en
     RAM, et faites de ce fichier de macros le fichier par défaut
     (c'est-à-dire compilez-le avec le compilateur de SMW) ; ainsi,
     quitter et redémarrer AppleWorks ne prendra que quelques
     secondes.

*****************************************************************

                    Dimitri Geystor - Août 1988

-----------------------------------------------------------------
start

<Ctrl-D>:<all><oa-P><rtn>4<rtn><msg>' Donnez le nom d'accès complet du
   fichier TEXT à créer, puis RTN '!

                    Sauve le ficher sous forme TEXT sur disque.

Note : Vérifiez que sur votre configuration l'option "Imprimer un fich
  ier texte (ASCII) sur disque" correspond bien à l'option n° 4 . Sin
  on, mettez le chiffre approprié dans la macro (cela dépend du nombr
  e d'imprimantes que vous avez défini).

-----------------------------------------------------------------

<Ctrl-C>:<all><oa-Q><esc><rtn>3<rtn>2<rtn><msg>"_______Donnez le nom
  d'accès de votre programme TEXT"!

                    Charge le fichier TEXT sous forme AWP

-----------------------------------------------------------------

---Cette macro doit être actionnée en premier : elle fait la "mise en
   page standard" du programme :

<tab>:<all><zoom><oa-Z><oa-1><oa-M>R<right><right><right><right><right
   ><right><right>F<esc><oa-Z><oa-0>IN<rtn>8<rtn>mg<rtn>,5<rtn>md<rtn
   > 0<rtn>mh<rtn>,5<rtn>mb<rtn>,5<rtn><esc><oa-9><rtn>§<rtn>§<rtn>!

-----------------------------------------------------------------

---Ces deux macros "formatent" le programme :

---"Aération" en séparant chaque numéro de ligne par 5 tirets
<ctrl-A>:<all><insert><sa-rtn><left><sa-1><right><oa-right>-----<rtn><
   rpt>!

1:<all><if>§<oa-1><zoom><stop>!

---"Ventilation" en mettant une seule instruction par ligne
<ctrl-V>:<all><insert><right><sa-2><sa-3><rpt>!

2:<all><if>:<rtn><spc><spc><spc><spc><spc><spc><spc><right>!

3:<all><if>§<oa-1><stop>!

-----------------------------------------------------------------

---Cette macro "justifie à droite" la colonne des numéros de ligne :

j:<all><sa-8><down><rpt>!

8:<all><oa-tab><sa-1><right><ifnot><spc><var=>4<sa-9><oa-tab><sa-ù>!

9:<all><right><ifnot><spc><decvar><rpt>!
```

La commande EXEC

Pour mieux comprendre cette commande, dites vous simplement que :

• EXECuter un fichier TXT conduit exactement au même résultat que de taper au clavier le contenu exact de ce fichier pour le mettre en mémoire.

• Si vous EXECutez un fichier TXT (ASCII) qui contient les lignes d'un programme en Basic, c'est comme si vous tapiez ce programme au clavier (sauf que seul le caractère "§" apparaît pour chaque ligne - et les éventuels messages d'erreur si la ligne est incorrecte aux yeux de Basic System). Mais le programme est bien 'entré' en mémoire : il suffit de faire LIST pour s'en convaincre.

• C'est pour cela que deux EXEC successifs mettent deux programmes bout à bout (si les numéros ne se chevauchent pas), exactement comme si vous continuiez à entrer au clavier la suite du même programme.

```
ù:<all><insert><varnot>0<spc><decvar><rpt>!

-----------------------------------------------------------------------

---Cette macro remet le texte du programme en format "standard" : chaq
   ue ligne numérotée séparée par un seul "retout chariot" :

<ctrl-R>:<all><oa-l><sa-*>!          Supprime l'aération (tirets)

*:<all><sa-rtn><oa-right><sa-5><sa-4><rpt>!

4:<all><if>-<right><if>-<sa-rtn><del><del><del><del><del><del>!

5:<all><if>$<oa-l><sa-ctrl-E>!       Supprime la "ventilation"

<ctrl-E>:<all><sa-rtn><oa-right><sa-6><oa-right><sa-7><rpt>!

6:<all><if>$<oa-l><zoom><stop>!

7:<all><if>:<del><del><del><del><del><del><del><del><del>!

-----------------------------------------------------------------------

---Cette macro supprime toutes les instructions REM (avant de l'action
   ner, mettre le programme en format "ventilé") :

<ctrl-N>:<all><oa-L><rtn><oa-Y><spc>rem<spc><rtn><rtn><oa-tab><oa-tab>
    <oa-E><tab>tab><tab><rtn><rpt>!

-----------------------------------------------------------------------

---Cette macro insère 5 tirets, pour "aérer" les lignes :

s:<all><rtn>-----<rtn>!

---Ces macros facilitent l'écriture en format "ventilé" :

::<all><tab><tab><rtn><tab>:<right>!            Instruction suivante

z:<all><oa-tab><oa-tab><oa-D>s<tab><tab><rtn>!  "Zapper la ligne"

b:<all><tab><tab><oa-C>D!                Récupérer la ligne "zappée"

-----------------------------------------------------------------------
```

sous la forme d'un fichier TEXT (procédure 1 et 2 ci-dessus). Supposons qu'il ait pour nom CHARGEUR.

Voici ce programme :

```
63500 d$ = chr$(4)
63506 input "Fichier : " xx$
63508 ?d$ "open" xx$
63510 ?d$ "write" xx$
63512 list 1,63499
63514 ?d$ "close"
63516 end
```

Pour transformer n'importe quel programme Basic Applesoft en fichier AWP, procéder comme suit

① À partir du Basic Applesoft, faire LOAD PROG (en supposant que PROG est le nom du programme en Basic).

② Aussitôt après, faire EXEC CHARGEUR. Une série de '§' s'impriment à l'écran : c'est normal, et signe que tout va bien. Lister : les deux programmes PROG et CHARGEUR, mis bout à bout, n'en font plus qu'un seul.

③ Faire RUN 63500. La partie 'Chargeur' est exécutée, et vous demande d'indiquer un nom de fichier : indiquez, par exemple, PROG.TXT.2 (ce sera le nom de de votre programme sous forme TXT).

④ Cataloguez le disque pour vérifier que PROG.TXT.2 y est bien.

⑤ Lancez AppleWorks, et créez un fichier traitement de textes "d'un fichier texte (ASCII)", en donnant le nom d'accès de PROG.TXT.2.

Le programme CHARGEUR est créé une fois pour toutes, et résidera sur la disquette où seront vos programmes en Basic (il est déjà sur la disquette Pom's 38).

Les opérations sont très simples, et peuvent être automatisées en ce qui concerne AppleWorks. Voici les deux macros qui permettront 'd'imprimer sur disque (Ascii)' et de créer un fichier AWP 'd'un fichier texte (ASCII)' :

```
<Ctrl-D>:<all><oa-P><rtn>4<rtn
   ><msg>' Donnez le nom d'accè
   s complet du fichier TEXT à
   créer, puis RTN '! Sauve le
   ficher sous forme TEXT sur d
   isque.
```

Note : vérifiez que sur votre configuration l'option "Imprimer un fichier texte (ASCII) sur disque" correspond bien à l'option n° 4. Sinon, mettez le chiffre approprié dans la macro (cela dépend du nombre d'imprimantes que vous avez défini).

```
<Ctrl-C>:<all><oa-Q><esc><rtn>
   3<rtn>2<rtn><msg>"________Do
   nnez le nom d'accès de votre
    programme TEXT"! Charge le
   fichier TEXT sous forme AWP
```

Ces deux macros sont incluses dans le fichier MACROS.BASIC.

Formater

Prenez un quelconque petit programme en Basic (la disquette Pom's contient EXEMPLE.PRG, qui fera bien l'affaire), et appliquez-lui la procédure B ci-dessus, point 1 à 4 inclusivement.

Bootez AppleWorks, chargez le fichier MACROS.BASIC, faites Ó-= pour le compiler, et actionnez la macro <sa-Ctrl-C>. Vous nommerez ESSAI.PRG votre fichier AWP.

L'opération qu'il faut faire aussitôt après est d'appliquer la macro <sa-tab>, qui fait plusieurs choses :

① Elle fixe une certaine 'mise en page' standard : tabulation, marges, longueur de papier pour l'impression

② Elle ajoute deux '§' en fin de fichier, qui serviront de 'drapeau' pour arrêter la boucle des macros de formatage. Quand vous ferez l'opération de transfert inverse (awp ➡ Basic), pensez à enlever ces drapeaux. Ils n'empêcheront cependant pas votre programme de toumer, en cas d'oubli.

Il est aussi recommandé d'actionner à ce stade la macro <sa-J>, qui justifie à droite tous les numéros des lignes d'instruction. Cette macro a un intérêt purement esthétique. Par la suite, il ne sera pas nécessaire de la 'défaire', cette justification ne gênant en rien la transformation du fichier AWP en fichier TXT, puis BAS.

```
---Macros diverses "sténographiques" :

----- CHR$ :
S:<all><insert>CHR$ (<msg>' Valeur ASCII, puis RTN '<input>)<right><ms
   g>""!

----- PRINT :
P:<all><insert>PRINT<right>!

----- DATA
D:<all><insert>DATA<right>!

----- HTAB
H:<all><insert>HTAB<right>!

----- VTAB
V:<all><insert>VTAB<right>!

----- IF ... THEN
I:<all><insert>IF <msg>' condition, puis RTN
'<input><right>THEN<right><msg>""!

----- PEEK ( )
E:<all><insert>PEEK (<msg>' Adresse, puis RTN '<input>)<right><msg>""!

----- POKE
K:<all><insert>POKE <msg>' Adresse, puis RTN '<input>,<msg>' Valeur, p
   uis RTN '<input><right><msg>""!

----- CHR$
$:<all><insert>CHR$ (<msg>' Valeur ASCII, puis RTN '<input>)<right><ms
   g>""!

----- MID$
M:<all><insert>MID$ (<msg>' Chaine, puis RTN '<input>,<msg>' Début ? p
   uis RTN '<input>,<msg>' Longueur, puis RTN '<input>)<right><msg>""!

----- LEFT$
L:<all><insert>LEFT$ (<msg>' Chaine, puis RTN '<input>,<msg>' Longueur
   , puis RTN '<input>)<right><msg>""!

----- RIGHT$
R:<all><insert>RIGHT$ (<msg>' Chaine, puis RTN '<input>,<msg>' Longueu
   r, puis RTN '<input>)<right><msg>""!

----- PEEK ( ) + 256 * PEEK ( )
W:<all><insert>PEEK (<msg>' Adresse 1, puis RTN '<input>)<right>+<righ
   t>256<right>*<right>PEEK (<msg>' Adresse 2, puis RTN '<input>)<msg>
   ""!

----- VAL
A:<all><insert>VAL (<msg>' Chaine, puis RTN '<input>)<right><msg>""!

----- GOTO
O:<all>GOTO<right>!

----- GOSUB
U:<all>GOSUB<right>!

-----------------------------------------------------------------------
END
```

Vous pouvez maintenant vous exercer à utiliser les macros <sa-Ctrl-A>, <sa-Ctrl-V> et <sa-Ctrl-R>. La plus lente à s'exécuter est la macro <sa-Ctrl-V> : que voulez-vous, ce n'est pas du langage-machine, et la macro lit tous les caractères pour repérer le séparateur ":" qui lui indiquera qu'il faut aller à la ligne.

Les macros <sa-Ctrl-A> et <sa-Ctrl-V> agissent à partir de la position du curseur, la macro <sa-Ctrl-R> commence toujours en début de fichier.

Si vous pensez vous contenter de la

seule 'ventilation', faites <sa-Ctrl-V> seulement : le formatage sera bien plus rapide, car il y aura moins de lignes à lire.

Les macros <sa-Ctrl-A> et <sa-Ctrl-R> travaillent plus vite car elles ne testent pas tout le texte caractère par caractère.

La macro <sa-Ctrl-R> fait un ménage général, et 'reconstruit' le programme sous une forme appropriée pour le transfert vers le Basic Applesoft. Elle commence par éliminer les tirets, puis revient au début et 'compacte' les instructions, en ne laissant qu'un seul retour chariot par ligne de programme.

Le fichier MACROS.BASIC contient une option de formatage supplémentaire : elle permet, une fois qu'un programme a été mis au point, d'en enlever toutes les remarques, pour une une exécution plus rapide. Certaines précautions sont à respecter :

☞ Cette macro, <sa-Ctrl-N>, doit s'appliquer à un programme sous sa forme 'ventilée' (chaque instruction à la ligne).

☞ Cette macro a été prévue (pour accélérer son fonctionnement) sans test de fin de fichier. À vous de surveiller sa progression et, quand la fin du fichier a été dépassée, de faire ESC pour arrêter la macro. Il faut aussi, après la macro, 'nettoyer' les éventuels caractères parasites qui auront peut-être eu le temps de s'imprimer à la suite du programme.

☞ Chaque effacement de REM ne porte que sur une ligne : si une remarque est trop longue et déborde sur la ligne suivante, cette partie ne sera pas effacée. Si vous tenez absolument à un commentaire abondant, répartissez-le sur plusieurs REMs.

☞ Méfiez-vous des GOTO et GOSUB qui renvoient à des sous-routines qui commencent par une REM ! Le numéro de ligne appelé ne doit pas être celui de la REM qui va disparaître.

Une fois assimilées ces opérations de base, vous êtes prêts à modifier des programmes existants, ou à écrire de nouveaux programmes.

Les macros du fichier MACROS.BASIC vous y aideront.

Comment modifier un programme

Vous disposez de toutes les fonctions d'édition d'AppleWorks et de SuperMacroWorks : le confort des déplacements rapides avec la souris, les fonctions de recherche et de remplacement, les effacements, les couper/coller, etc.

À propos de souris, je rappelle qu'elle vous donne deux façons de faire défiler les écrans : en continu, en amorçant le mouvement et en maintenant le bouton enfoncé, et écran par écran. C'est le défilement le plus rapide : gardez enfoncée la touche ⇧, et 'traînez' doucement la souris. J'ai bien dit doucement.

Pour le 'contexte' de l'édition, c'est une affaire de goût :

☞ Vous pouvez choisir de travailler sur le texte de programme 'non formaté', sans précautions particulières, sinon de veiller à avoir un seul 'retour chariot' à la fin de chaque ligne numérotée, ni plus ni moins. Faites <oa-Z> pour contrôler.

☞ Vous pouvez travailler sur le texte de programme 'aéré' : les tirets sont des repères visuels bien utiles. Si par mégarde on supprime un retour-chariot en fin de ligne, les tirets se décalent aussitôt. Cela saute aux yeux, et il suffit de faire RTN pour les remettre à leur place. Il faut qu'au moment de la 'reconstruction' avec <sa-Ctrl-R> tous les tirets soient en début de ligne.

☞ Vous pouvez enfin travailler sur le texte 'ventilé'. La syntaxe est différente : pour respecter la cohérence, il faut non seulement aller à la ligne après chaque instruction, mais respecter la tabulation des séparateurs ":".

Le fichier MACROS.BASIC contient deux commandes pour vous faciliter le travail :

① <sa-:> : quand elle est actionnée n'importe-où dans une ligne, cette macro insère automatiquement une nouvelle ligne, fait la tabulation, imprime ":", et place le curseur. Il n'y a plus qu'à taper la nouvelle instruction. Quand celle-ci est écrite, on tape ⌘-: pour passer à l'instruction suivante.

② <sa-Z> 'zappe' une instruction (donc une ligne) entière. La ligne zappée est copiée dans le presse-papiers, d'où <sa-B> la récupère au besoin, en l'insérant après la ligne du curseur. Ces deux macros permettent donc un "couper/coller".

Comment écrire un nouveau programme

Créez le fichier de traitement de texte, et appliquez-lui aussitôt la macro <sa-tab>.

Pour écrire le programme, le format 'aéré' est visuellement le plus satisfaisant, mais nullement obligatoire :

☞ Écrivez le numéro de ligne, faites TAB, écrivez les instructions selon la syntaxe habituelle Applesoft. Pour passer à la ligne suivante, faites <sa-S> : cette macro va à la ligne en insérant cinq tirets.

☞ Si vous préférez écrire en 'format ventilé', c'est-à-dire une instruction par ligne, commencez la ligne normalement, puis utilisez la macro <sa-:> pour chaque nouvelle instruction. Pour passer à la ligne numérotée suivante, faites seulement RTN. Ce format est commode si votre programme doit écrire du texte à l'écran, car les instructions placées les unes sous les autres vous donnent une bonne idée de l'apparence future de l'écran (en 80 colonnes).

☞ Rien ne vous interdit d'écrire en 'mode mixte' : pour certaines lignes en format 'aéré' (ou même tout à fait ordinaire), pour d'autres en 'ventilant' instruction par instruction. Quand vous ferez la 'toilette' finale de votre texte, la macro <sa-Ctrl-R> du fichier MACROS.BASIC convertira un texte 'ventilé' au format normal, sans rien changer pour autant aux parties de

texte qui seront déjà en format 'normal'.

Autres macros du fichier MACROS.BASIC

En plus des trois macros <sa-:>, <sa-Z> et <sa-S> déjà mentionnées, le fichier MACROS.BASIC contient un certain nombre de macros destinées à vous faciliter l'écriture d'expressions répétitives du Basic Applesoft.

Je rappelle que les macros peuvent être tapées en utilisant indifféremment les lettres minuscules ou majuscules.

<sa-rtn> Va au prochain retour-chariot (fin de ligne ou fin d'instruction, selon le mode)

<sa-P> Écrit PRINT, avance d'un espace

<sa-D> Écrit DATA, avance d'un espace

<sa-H> Écrit HTAB, avance d'un espace

<sa-V> Écrit VTAB, avance d'un espace

<sa-I> Écrit IF, attend la condition, complète avec THEN, avance d'un espace

<sa-E> Écrit PEEK(, attend une valeur, referme la parenthèse et avance d'un espace

<sa-K> Écrit POKE, attend deux valeurs, avance d'un espace

<sa-$> Écrit CHR$(, attend la valeur Ascii, referme la parenthèse, avance d'un espace

<sa-M> Écrit MID$(, attend trois indications, referme la parenthèse, avance d'un espace

<sa-L> Écrit LEFT$(attend deux indications, referme la parenthèse, avance d'un espace

<sa-R> Écrit RIGHT$(attend deux indications, referme la parenthèse, avance d'un espace

<sa-W> Écrit un PEEK () + 256*peek () , en faisant deux pauses pour l'insertion des valeurs, avance d'un espace

<sa-A> Écrit VAL(, attend la chaîne, referme la parenthèse, avance d'un espace

<sa-O> Écrit GOTO, avance d'un espace

<sa-U> Écrit GOSUB, avance d'un espace

<sa-T> Écrit RETURN, avance d'un espace

<sa-G> Écrit STR$(, attend la valeur, ferme la parenthèse et avance d'un espace

<sa-Ctrl-G> Écrit BELL, avance d'un espace

<sa-F> Écrit FOR, attend une valeur, écrit =, attend une valeur, écrit TO, attend une valeur, avance d'un espace

<sa-N> Écrit NEXT, avance d'un espace

<sa-"> Ouvre des guillements, attend un texte, referme les guillemets

Cette liste n'est évidemment pas exhaustive, et chacun pourra la compléter à son gré.

Pour essayer votre programme, quatre opérations simples, qui ne vous prendront que quelques secondes :

① 'Recompaeter' le fichier AWP avec la macro <sa-Ctrl-R>, que vous pouvez actionner de n'importe quel endroit de ce fichier.

② Allez en fin de fichier, et supprimez les drapeaux "§".

③ Faites <sa-Ctrl-D>, et créez votre fichier PROG.TXT.

④ Quittez AppleWorks, allez en Basic, faites "EXEC PROG.TXT" (ou —PROG.TXT) et RUN.

Les programmes proposés par Pom's

Voici une table indiquant dans quel numéro de Pom's vous trouverez une description de nos programmes diffusés :

CLV_Pom's	31	programme d'enregistrement de CalvaCom
Éditeur Vidéotex	31	Éditeur d'écran Minitel (CP/M)
SuperMacroWorks	31	Macros & souris sur AppleWorks
InterPom's	31	Transmission de fichiers
Pom_Link 3.1	34 & 36	Serveur télématique complet
Beagle Compiler	35	Puissant compilateur Basic sous ProDOS
Big U	35	Ensemble d'utilitaires indispensables au Basic
ProSel	36 & 37	Sélecteur d'applications irremplaçable
Copy II Plus 8.3	37	Copieur pour Apple //
Copy II Mac	37	Copieur pour Macintosh
TimeOut	38	Ensemble d'applications sous AppleWorks

SHAREWARE COMMUNICATION SUR LA DISQUETTE 38 POUR APPLE //

Sur la disquette 5.25', vous trouverez **Kermit** v3.78
Sur la disquette 3.5' vous aurez également **Deluxe Disk Desintegrator** v4.2, **Binary Linking Utility** v2.27, le compacteur pour GS **Digipack** v1.0, Binary II, **Executioner** de Glen Bredon, **TalkIsCheap**, le compacteur **ProArc**.

L'ensemble avec documentation en anglais. Pom's n'assure aucune assistance pour ces programmes shareware.

AppleWorks :

A.-P. Gaspar

Impression sur 2 colonnes

Depuis qu'AppleWorks a vu le jour, l'impression d'un texte sur deux (ou plusieurs) colonnes est le rêve insaisissable que poursuivaient ses utilisateurs.

Il y a quelques mois 'l'expert en AppleWorks' Charles Rubin proposait, dans une revue américaine, 'sa' solution. En gros, il s'agissait d'utiliser le traitement de texte, la base de données et le tableur : on commençait dans le traitement de texte, on faisait le transfert dans le tableur via la base de données, on organisait le texte en deux colonnes dans le tableur, et on revenait au traitement de texte via le presse-papiers. Procédure lourde et compliquée, même en recourant à des macros pour le transfert. Son seul avantage : on voyait le texte sur deux colonnes dans le traitement de texte (et encore, à condition de ne pas dépasser 78 caractères par ligne, et de pas trop y toucher : rien ne se démolie plus traîtreusement que ce genre d'architecture !).

La réponse ne se fit pas attendre, dans le courrier des lecteurs de cette même revue, peu de temps après : pourquoi se compliquer la vie, écrivait ce lecteur, alors qu'il y a une solution beaucoup plus simple, à la condition de faire passer le papier deux fois de suite dans l'imprimante ! «On commence par formater le texte en une seule colonne sur la moitié gauche de la page, et on imprime. Puis on formate le même texte en une seule colonne sur la moitié droite de la page, on réintroduit le papier dans

Principes de fonctionnement

① choisir une série de pages sur un fichier tableur, ce qui permet de bénéficier d'un fonction rythmique (impression de la page courante + 2, + 3, etc.). Dans la version présentée ici, la grille des pages se constitue automatiquement (macro-Y) avec 2 colonnes : une colonne de 2 en 2 à partir de la page 0 (nommée ici 444 car AppleWorks refuse de trouver une page 0) et une colonne de 2 en 2 à partir de la page 1.

② lire successivement les cases du tableur, en faisant passer l'information sur un fichier intermédiaire de type texte qui permet de récupérer le nombre et non la fonction.

③ faire une recherche de la page concernée dans le fichier texte courant.

④ imprimer de façon sélective la page concernée.

⑤ reconnaître la fin du fichier.

⑥ revenir au début du fichier pour l'impression des pages de la deuxième colonne de la grille.

Macros-commandes 'IMP.2.COLONNES'

```
START

Sous-routines:

L:<all><oa-Q>3<Rtn><oa-P>B<Rtn>2<Rtn><Rtn><Down><oa-Q>2<Rtn><oa-Y
   ><oa- C>D<oa-Right><read><read><read><oa-1><oa-Q>1<Rtn>!

ATTENTION : dans la macro L ci-dessus, le "2" après <oa-P>B<rtn>
   correspond à l'option "Imprimer sur le presse-papiers (pour le
    Traitement de texte). Ceci peut varier d'une configuration à
   l'autre : vérifiez, et mettez la valeur qui convient.

M:<all><sa-L><oa-L>P<sa-0><rtn><rtn><down><down><ifnot>*<up><up><
   oa-P> C1<rtn><rtn><esc><rpt>!

N:<all><sa-L><oa-L>P<sa-0><rtn>0<rtn><oa-P>C1<rtn><rtn><esc>!

F:<all><zoom><oa-Z><oa-1><oa-O>mg<rtn>0,1<rtn>md<rtn>4,1<rtn>pc<r
   tn>no <rtn>1<rtn><esc><oa-K>1<rtn>!

G:<all><zoom><oa-Z><oa-1><oa-O>mg<rtn>4,1<rtn>md<rtn>0,1<rtn><esc
   ><oa- K>1<rtn>!

H:<all><zoom><oa-Z><oa-1><oa-O>mg<rtn>1<rtn>md<rtn>0,5<rtn>pc<rtn
```

l'imprimante, et on imprime une deuxième fois, mais en commençant par la deuxième page».

Minute, me direz-vous, c'est astucieux, mais cet œuf de Christophe Colomb me fait jeter une page sur deux au panier ! Certes, mais on ne fait pas d'omelette, etc. Ou alors, il faudrait, lors de la première passe, n'imprimer que les pages impaires, et lors de la deuxième n'imprimer que les pages paires.

Mais au fait, AppleWorks a toutes les commandes qu'il faut : ⌘-K pour numéroter les pages, ⌘-L pour localiser les pages, l'option d'impression PC (pause à chaque page) qui arrête l'impression le temps de faire avancer le curseur d'une page sur deux. Il ne manque que l'esclave pour 'tourner la manivelle'. Mais justement, AppleWorks a tout cela, et même mieux, avec SuperMacroWorks. Car SuperMacroWorks sait compter, et on peut parfaitement le charger de sauter d'abord toutes les pages paires, ensuite toutes les pages impaires. On aurait pu, pour cela, utiliser un petit compteur du genre de ceux que je vous ai présentés dans les numéros précédents.

Dans l'illustration qui va suivre, un lecteur de Pom's, ***Alexandre-Pierre Gaspar****, a choisi de faire faire le décompte par le tableur.*

Tout d'abord, on construit dans le tableur deux colonnes de chiffres qui correspondront aux pages, paires et impaires. La macro lira les cellules successives (non pas directement, mais en les imprimant dans un fichier de traitement de texte, pour lire le nombre et non la formule - tiens, tiens, ça ne vous rappelle rien ? Voici une méthode de plus pour ce genre de transfert). Puis elle mettra le résultat de la lecture dans la macro 0 (zéro), ira dans le traitement de texte, localisera la page enregistrée dans la macro 0, etc.

Si le mécanisme vous intéresse, épluchez les macros d'A.-P. Gaspar, c'est très instructif.

```
   >no<rtn>1<rtn><esc><oa-K>1<rtn>!

I:<all><zoom><oa-Z><oa-1><oa-O>mg<rtn>0,5<rtn>md<rtn>1<rtn><esc><
   oa-K> 1<rtn>!

X:<awp><oa-9><oa-O>JG<rtn>np<rtn><esc>*<rtn><oa-●>np<rtn><esc>*<r
   tn>!

Y:<all><rtn>3<rtn><rtn>INTERFACE<rtn><esc><Rtn>5<Rtn><Rtn>GRILLE<
   Rtn>4 44<Rtn><Right>1<Rtn><Down><Left>2<Rtn><Right>3<Rtn><Down
   ><Left>+<Up >+2<Rtn><Right>+<Up>+2<Rtn><Left><oa-c><Rtn><Right
   ><Rtn><Down>.<oa- Down><oa-Down><oa-Down><oa-Down><oa-Down><oa
   -Down><oa-Down><oa-Down><oa-Down><oa-Down><oa-Down><Down><Down
   ><Rtn><Right><Rtn><Right><Rt n><oa-1><oa-O>ph<rtn><esc><esc>!

E:<all>4<rtn><down><rtn>3<rtn>O4<rtn><down><rtn>3<rtn>O!
Macros générales :

F:<all><esc><sa-Y><oa-●>1<rtn><sa-X><oa-1><sa-F><sa-N><sa-M><esc>
   <msg> 'Remettre le listing au début.'<key><oa-Q>3<rtn><oa-1><t
   ab><esc><oa-Q>1<rtn><oa-1><sa-G><sa -N><sa-M><oa-9><oa-E><up><
   up><up><up><up><up><rtn><oa-1><esc><sa-E> !

Q:<all><esc><sa-Y><oa-●>1<rtn><sa-X><oa-1><sa-H><sa-N><sa-M><esc>
   <msg> 'Remettre le listing au début et au recto.'<key><oa-Q>3<
   rtn><oa-1><tab><esc><oa-Q>1<rtn><oa-1><sa-I><sa -N><sa-M><oa-9
   ><oa-E><up><up><up><up><up><up><rtn><oa-1><esc><sa-E> !

END
```

Les Macros-commandes

- **L** Lecture d'une case de la grille des pages dans la macro-0
- **M** Macro répétitive de recherche et d'impression des pages
- **N** Impression de la première page de chaque colonne de la grille des pages
- **F** Formatage du fichier pour l'impression de la première colonne: définition des marges gauche (0,1 pouce) et droite (4,1 pouces) en tête du fichier texte
- **G** Formatage du fichier pour l'impression de la deuxième colonne: définition des marges gauche (4,1 pouces) et droite (●.1 pouce) en tête du fichier texte
- **H** Formatage du fichier pour l'impression recto : définition des marges gauche (1 pouce) et droite (0,5 pouce) en tête du fichier texte
- **I** Formatage du fichier pour l'impression verso: définition des marges gauche (0,5 pouce) et droite (1 pouce) en tête du fichier texte
- **X** Positionnement des marques de fin de fichier

Si la seule chose qui vous préoccupe est le résultat, qu'à cela ne tienne, faites tourner la macro, elle marche à merveille (il y a juste une chose à vérifier : à partir d'un fichier tableur, dans votre configuration de travail, regardez quel est le numéro de l'option "imprimer sur le presse-papiers (pour le Traitement de textes)". Ce numéro est 2 dans la macro <sa-L>, mais il peut être 3 ou 4 selon le nombre d'imprimantes que vous avez définies simultanément. Corrigez si nécessaire.

Votre fichier à imprimer peut contenir des options de mise en page (CI, LP, IN, JT, etc.) : c'est le grand intérêt de cette méthode, vous avez un contrôle complet sur la présentation du texte (contrairement à la solution de Charles Rubin). Une seule chose à respecter : ou bien vous évitez complètement toute indication des marges gauche et droite (la macro le fera pour vous, le moment venu), ou à l'extrême rigueur vous pouvez indiquer une marge de gauche en toute première position

parmi toutes les options d'impression, et une marge de droite en deuxième position.

Ah oui, j'oubliais, pour les perfectionnistes : après avoir terminé d'écrire votre fichier, faites ⌘-K, et regardez si par hasard il n'y a pas de 'nouvelle pages' qui commencent par une ligne vide (ça peut arriver à un changement de paragraphe). En deux colonnes, ça saute aux yeux et c'est disgracieux ; éliminez la ligne vide avant d'imprimer (et, bien sûr, vérifiez à nouveau avec ⌘-K une fois de plus, car vous venez de tout décaler).

DG

Y Création de la GRILLE (tableur) des pages à imprimer et de l'INTERFACE (fichier texte) de lecture des cases de la grille

E Effacement final des fichiers INTERFACE et GRILLE.

Mise en œuvre

① compiler les macros impression.

② charger le fichier texte à imprimer (il doit se trouver seul sur le bureau).

③ à partir de ce fichier, tapez :

☞ ⌘-P (bi-colonnes)

☞ ⌘-Q (recto-verso)

④ quand la procédure vous le demandera, remettez le papier au début, inversé (recto-verso) ou non (bi-colonnes), et appuyez sur une touche.

Note : les seules définitions de marge pouvant apparaître dans le fichier pour rester compatible avec la macro bi-colonnes sont : marge gauche en première ligne de fichier; marge droite en deuxième ligne de fichier.

Nouveau look AppleWorks, suite

E. Weyland

«Pom's est prêt à publier vos patches les plus réussis. Nul n'étant infaillible, mes modifications sont peut-être incomplètes, ou peuvent engendrer des pépins. Nous comptons sur vous, lecteurs, pour nous le faire savoir, si possible dans le Format Préféré de Pom's (FPP : deux colonnes. À gauche, la question. À droite, la réponse). Merci d'avance.»

Voilà comment se concluait mon article dans Pom's 36. Eh bien, vous avez répondu, en masse, et le FPP, vous connaissez. Je ne peux pas citer les noms de tous : c'est la communauté des lecteurs de Pom's, tout simplement, qui a relevé et corrigé coquilles et bugs, et nous sommes tous des "patchmaniaques", selon la formule de Randy Brandt. Les éditeurs de blocs et le désassemblage n'ont plus de secret pour vous.

Vous trouverez donc ci-dessous les rectifications nécessaires aux tableaux publiés dans Pom's 37 (il y avait quelques coquilles, et un bug : les nouveaux taquets de tabulation n'étaient pas reconnus par la touche TAB), ainsi qu'une série de patches de J.L. Rénié qui remplacent les flèches dessinées par "-->" par des icônes Mousetext.

J'ai compris la leçon : ce qu'il faut quand on veut un patch à AppleWorks, c'est 'lâcher la meute' des lecteurs de Pom's.

On recherche

① AppleWorks 1.4 a le mérite d'envoyer systématiquement à l'imprimante un 'espace arrière' (backspace, Contrôle-H) après chaque '^' et chaque '~', ce qui nous permet d'imprimer à la française. Bravo. Mais il le fait aussi quand on imprime un fichier texte sur disque, ce qui peut

Vous trouverez donc aussi ci-dessous une première liste de 'demandes de patches'. Envoyez-nous vous aussi vos demandes, et, si vous le pouvez, vos réponses.

AppleWorks 2.0 : corrections

BLOC	OCTET	CHAINE	REMPLACER PAR	SITUATION	COMMENTAIRES
$ 83 0131	$ 03	60 3D 30 30 3D 3D 7C A5	60 30 3D 3D 3D 3D 8A A5	SEG.M1 Traitement de textes	En-tête des fichiers : taquets de tabulation
$ 92 D146	$ B6	BD 11 7C C9 7C D0 F2	BD 11 7C C9 8A D0 F2	SEG.M1 Traitement de textes.	Fonctionnement de la touche TAB
$ 92 D146	$1F9	BD 11 7C C9 7C D0 F4	BD 11 7C C9 8A D0 F4		
$103 0259	$173	A9 5F 20 84 11 AE 9C	A9 93 20 84 11 AE 9C	SEG.M1	
$104 0260	$ 00	60 04 7C 06 09 7C 02 5F 5F	60 04 9F 06 09 94 02 5F 5F	SEG.M1	

être gênant. Si quelqu'un pouvait repérer le passage concerné, on peut désormais, avec TimeOut et UltraMacros, faire des patches **provisoires**, des **patches mémoire**. Du coup, on pourrait peut-être envisager aussi d'introduire cette caractéristique dans AppleWorks 2.0.

② En base de données, il réapparaît encore une fenêtre avec ces maudits "!" quand on veut effacer un format de rapport. Si quelqu'un pouvait la trouver...

③ Comment modifier la ligne haute de "==========" et la ligne basse de "-------" ? C'est facile je crois, mais un patch Mousetext peut créer la confusion entre les écrans traitement de textes et les écrans de menus.

④ Patch pour les "!" de SuperMacroWorks et UltraMacros (boot). Un peu de désassemblage est nécessaire ici, l'un et l'autre se chargeant avant AppleWorks.

P.S. 1 - Le patch à AppleWorks 2.0 qui remplace, en traitement de textes, le damier par le symbole Mousetext <Return>, publié dans Pom's 35 page 63, empêche UltraMacros de trouver les Return dans sa macro <find> (⌘-Return). On peut soit refaire une macro ⌘⇧-Return avec <peek $10F5>, soit patcher UltraMacros comme le propose Randy Brandt lui-même :

```
BLOAD ULTRA.SYSTEM, T$FF,
    A8192.J
POKE ((PEEK (8197) + 256 *
    PEEK (8198)) + 8195), 77.J
BSAVE ULTRA.SYSTEM, TSYS,
    A8192.J
```

P.S. 2 - L'incapacité d'UltraMacros (à la différence de SuperMacroWorks) à afficher les caractères Mousetext produit des résultats curieux avec mes patches pour les menus dans la macro <msg>. Deux solutions : soit vous ne faîtes pas le patch en bloc $103 — octet $11E, soit vous donnez à la macro <msg> des messages de toute la longueur de la ligne de traits du bas.

AppleWorks 1.4 : corrections

BLOC	OCTET	ANCIENNE CHAINE	REMPLACER PAR	SITUATION	COMMENTAIRES
$ 83 0131	$ D7	60 30 30 3D 30 3D 3D 21 A5	60 30 30 30 3D 30 3D 8A A5	SEG.M1 Traitement de textes	En-tête des fichiers : taquets de tabulation
$ 92 D146	$ 26	B0 11 7D C9 21 D0 F2	BD 11 7D C9 8A D0 F2	SEG.M1 Traitement de textes.	Fonctionnement de la touche TAB
$ 92 D146	$169	8D 11 7D C9 21 00 F4	80 11 7D C9 8A 00 F4		
$ A0 D160	$1E3	D0 04 A9 21 D0 06	D0 04 A9 8A D0 06	SEG.M1 Traitement de textes	Modification des toquets de tabulation
$100 D256	$173	A9 SF 20 84 11 AE 9C	A9 93 20 84 11 AE 9C	SEG.M1	
$101 D257	$ 55	60 04 21 06 09 21 02 5F 5F	60 04 9F 06 D9 94 02 5F 5F	SEG.M1	

AppleWorks 1.4 : flèches Mousetext

Patches de Jean-Louis Rénié, que les vols de palombes ont inspiré pour améliorer les flèches d'AppleWorks.

BLOC	OCTET	ANCIENNE CHAINE	REMPLACER PAR	SITUATION	COMMENTAIRES
$00E D014	$083	60 04 20 20 20 3E 01 02	60 04 20 93 93 95 01 02	APLWORKS. SYSTEM (Menus)	Flèche sur le n° choisi.
$04B D075	$DA2	1D 2D 2D 3E 2D 6F 72 20 3C 2D 2D 20 20	1D 93 93 95 20 6F 7S 20 88 93 93 20 20	SEG.MO	change organisation écran OA-X + correction du Bug "or"
$04B D075	$0E0	73 24 2D 2D 3E 20 81 20 20 3C 2D 2D 20 20	23 24 93 93 95 20 81 20 20 88 93 93 20 20	Base	
$04B DD75	$1DD	0A 3C 2D 20 20 2D 50 6C	0A 88 93 93 93 20 50 6C	de	ligne bas <-- Plus
$04B 0075	$1ED	73 20 2D 2D 2D 3E 11 49	73 24 93 93 93 95 11 49	Données	ligne bas Plus -->
$061 0097	$148	72 74 20 2D 2D 3E 20 6F	72 74 20 93 93 9S 20 6F	SEG.M1 B d D	Menu Format Rapport
$061 D097	$150	75 20 3C 2D 2D 20 20 44	75 20 88 93 93 20 20 44	SEG.M1 B d D	Menu Format Rapport
$061 D097	$190	27 2D 2D 3E 20 81 20 3C 2D 2D 20 20	27 93 93 95 20 81 20 88 93 93 20 20	SEG.M1 B d D	Menu Format Rapport
$062 0098	$1A3	6E 0A 3C 20 20 20 20 50	6E 0A 88 93 93 93 20 50	SEG.M1	Format Rapport Ligne bas Plus --> et <-- Plus
$062 D098	$1B3	75 73 20 20 20 20 3E 11	75 73 20 93 93 93 95 11	B d D	
$069 D105	$110	2D 2D 3E 20 20 2D 3C 2D 2D 20	93 93 95 20 20 20 88 93 93 20	SEG.M1 B d D	Ride format rapport BdD
$105 D261	$0E7	60 03 20 20 3E A9 00	60 03 93 93 95 A9 00	SEG.M1 Menus	Menu disques, Sélection des fichiers
$11F D287	$149	60 03 2D 20 3E 02 20	60 03 93 93 95 02 20	SEG.M1 Menus	Idem pour le bureau

Jean-Luc Bazanegue

Un accessoire nostalgique : PR#6

De nombreux lecteurs pratiquant l'assembleur 68000 désiraient avoir un 'noyau' d'accessoire simple — mais complet — leur permettant de construire facilement leurs propres accessoires. Voici donc ce noyau, quelque peu agrémenté puisqu'il s'agit d'un accessoire permettant d'éjecter les disquettes — après les avoir mises à jour — et de 'redémarrer' le Macintosh.

Nous avons baptisé cet accessoire 'PR#6', ce qui ne manquera pas d'intriguer les utilisateurs de Macintosh qui ne sont pas passés par l'Apple // ; il convient donc de dire que PR#6 est la méthode — quasi-légendaire — utilisée par les Applemaniaques pour 'rebooter' leur ordinateur.

Le fait que cet accessoire soit avant tout un 'noyau' laisse apparaître des branchements qui ne sont pas utilisés ; c'est à ces emplacements que seront placées vos routines d'édition, de gestion du curseur, etc.

Nous n'avons pas ici placé les routines de Couper, Coller... qui ne sont employées que par les accessoires qui gèrent du texte ou du graphisme. Vous pouvez en revanche utiliser de telles routines publiées dans les précédents numéros de Pom's (Kruptos, Ad litteram...).

Fichier 'PR6.Job'

```
Asm     PR6.Asm     Exec            Edit
Link    PR6.Link    Font/DA Mover   Edit
```

Fichier 'PR6.Link'

```
]
/NoStrip
/Resources
PR6
/output Accessoire "PR#6"
/Type 'DFIL' 'DMOV'
$
```

Fichier 'PR6.Asm'

```
RESOURCE 'DRVR' 29 'PR#6'
INCLUDE         FSEqu.Txt
INCLUDE         Traps.Txt
INCLUDE         SysEqu.D
INCLUDE         ToolEqu.D
INCLUDE         QuickEqu.D
INCLUDE         PackMacs.Txt

HandleCtl       EQU     WindowSize
HandleControle  EQU     HandleCtl+4
TamponIO        EQU     HandleControle+4
TailleTampon    EQU     TamponIO+100

Oui             EQU     $100

Base
    DC          $400
    DC          0
    DC          $16A
    DC          0
    DC          Ouverture-Base
    DC          Status-Base
    DC          Controle-Base
    DC          Status-Base
    DC          Fermeture-Base
```

```
Titre
    DC.B        4,'PR#6',0

; Ouverture de l'accessoire

Ouverture
    MOVEM.L     D3-D7/A1-A4,-(SP)
    MOVEA.L     A1,A4
    TST.L       dCtlWindow(A4)
    BNE         Status4
    SUBQ.L      #4,SP
    MOVE.L      SP,-(SP)
    _GetPort
    MOVE.L      #TailleTampon,D0
    _NewPtr,clear
    TST         D0
    BEQ.S       MemoireOK
    MOVE        #7,-(SP)
    _SysBeep
    BRA         Status2

MemoireOK
    MOVEA.L     A0,A3
    SUBQ.L      #4,SP
    MOVE.L      A3,-(SP)
    PEA         RectFenetre
    PEA         Titre
    MOVE        #Oui,-(SP)
    MOVE        #noGrowDocProc,-(SP)
    MOVEQ.L     #-1,D0
    MOVE.L      D0,-(SP)
    MOVE        #Oui,-(SP)
    CLR.L       -(SP)
    _NewWindow
    _SetPort
    MOVE.L      A3,dCtlWindow(A4)
    MOVE.L      DctlRefNum(A4),WindowKind(A3)

; Affichage du bouton

    SUBQ.L      #4,SP
    MOVE.L      A3,-(SP)
    PEA         RectBouton
    PEA         TitreBouton
    MOVE        #Oui,-(SP)
    CLR.L       -(SP)
    CLR.L       -(SP)
    CLR.L       -(SP)
    _NewControl
    MOVE.L      (SP)+,HandleCtl(A3)
    _InitCursor
Status2
    _SetPort
Status4
    MOVE.L      dCtlWindow(A4),A3
Status3
    MOVEM.L     (SP)+,D3-D7/A1-A4
Status
    MOVEQ       #0,D0
    RTS

; Fermeture de l'accessoire

Fermeture

    MOVEM.L     A3-A4,-(SP)
    MOVEA.L     A1,A4
    MOVEA.L     dCtlWindow(A4),A3
    MOVE.L      A3,-(SP)
    _DisposWindow
    CLR.L       dCtlWindow(A4)
    MOVEA.L     A4,A1
    MOVEM.L     (SP)+,A3-A4
    BRA         Status

; Routine de contrôle

Controle
    MOVEM.L     D3-D7/A1-A4,-(SP)
    MOVEA.L     A1,A4
    MOVEA.L     A0,A2
    MOVEA.L     dCtlWindow(A4),A3
    MOVE.L      A3,-(SP)
    _SetPort
    MOVE        CSCode(A2),D0
    CMPI        #accEvent,D0
    BEQ.S       Evenement
    CMPI        #accCursor,D0
    BEQ         Curseur
    CMPI        #accCut,D0
    BEQ         Couper
    CMPI        #accCopy,D0
    BEQ         Copier
    CMPI        #accPaste,D0
    BEQ         Coller
    CMPI        #accClear,D0
    BEQ         Effacer
    BRA         Status3

; Aiguillage des événements

Evenement
    MOVEA.L     CSParam(A2),A2
    MOVE        EvtNum(A2),D0
    CMPI        #mButDwnEvt,D0
    BEQ.S       Contenu
    CMPI        #keyDwnEvt,D0
    BEQ         Touche
    CMPI        #autoKeyEvt,D0
    BEQ         Touche
    CMPI        #updatEvt,D0
    BEQ         MiseJour
    CMPI        #activateEvt,D0
    BEQ         Active
    BRA         Status3

; Dans la fenêtre de l'accessoire

Contenu
    PEA         EvtMouse(A2)
    _GlobalToLocal
```

```
    CLR         -(SP)
    MOVE.L      EvtMouse(A2),-(SP)
    MOVE.L      A3,-(SP)
    PEA         HandleControle(A3)
    _FindControl
    TST         (SP)+
    BEQ         Status3
    CLR         -(SP)
    MOVE.L      HandleControle(A3),-(SP)
    MOVE.L      EvtMouse(A2),-(SP)
    CLR.L       -(SP)
    _TrackControl
    TST         (SP)+
    BEQ         Status3
    BRA         Bouton

Touche
    BRA         Status3

; Routine de mise à jour

MiseJour
    MOVE.L      A3,-(SP)
    _BeginUpdate
    BSR         PoliceChicago
    BSR         AffichePoms
    MOVE.L      #$00100030,-(SP)
    _MoveTo
    PEA         Chaine
    _DrawString
    MOVE.L      A3,-(SP)
    _DrawControls
    MOVE.L      A3,-(SP)
    _EndUpdate
    BRA         Status3

PoliceChicago
    CLR         -(SP)
    _TextFont
    MOVE        #12,-(SP)
    _TextSize
    RTS

AffichePoms
Tampon          SET         -bitmaprec
    LINK        A6,#Tampon
    MOVEM.L     A0-A4/D0-D2,-(SP)
    LEA         Poms,A0
    MOVE.L      A0,Tampon+baseAddr(A6)
    MOVE        #4,Tampon+rowBytes(A6)
    CLR.L       Tampon+bounds(A6)
    MOVE.L      #$200020,
                Tampon+bounds+bottom(A6)
    PEA         Tampon+baseAddr(A6)
    MOVEA.L     A3,A0
    ADDQ.L      #2,A0
    MOVE.L      A0,-(SP)
    PEA         Tampon+bounds(A6)
    PEA         RectPoms
    CLR         -(SP)
```

```
    CLR.L       -(SP)
    _CopyBits
    MOVEM.L     (SP)+,A0-A4/D0-D2
    UNLK        A6
    RTS

; Pour faire un 'noyau' complet

Couper
    BRA         Status3
Copier
    BRA         Status3
Coller
    BRA         Status3
Effacer
    BRA         Status3
Curseur
    BRA         Status3
Active
    BRA         Status3

Bouton
    MOVE        #1,ioVDrvNum(A0)
    _Eject
    MOVE        #2,ioVDrvNum(A0)
    _Eject
    JMP         $40000A

RectFenetre     DC          41,4,88,242
RectBouton      DC          24,162,40,216
RectPoms        DC          8,8,40,40

Chaine          DC.B        24,'Voulez-vous
                            redémarrer ?',0
TitreBouton     DC.B        3,'Oui'
                .Align      2

Poms            DC.L        $00000000,$00000050
                DC.L        $00000080,$00001100
                DC.L        $000008AA,$00000155
                DC.L        $000000AA,$00001401
                DC.L        $00002A02,$00004514
                DC.L        $0000A280,$00015140
                DC.L        $000228A0,$00051440
                DC.L        $00028A00,$00014560
                DC.L        $0028A2A0,$00545140
                DC.L        $008A2980,$00453D7C
                DC.L        $00A2FFD6,$1451FEAB
                DC.L        $2A28FD45,$4515FAA3
                DC.L        $A20AFD45,$5100FAA3
                DC.L        $2A007D46,$14007AAA
                DC.L        $0A003D54,$05001AA8
                DC.L        $02800DD0,$01000660
```

Puzzle

Marc Girati

Puzzle est une application ludique des possibilités graphiques du Macintosh. Entièrement écrite en assembleur 68000 (les amateurs apprécieront), l'application est d'un emploi très simple, ce qui a entre autres avantages celui de limiter la taille du mode d'emploi à quelques lignes.

Il est en effet simplement nécessaire d'ouvrir un document 'Paint' (MacPaint, FullPaint...) ; Puzzle en extrait la partie supérieure gauche, l'affiche, puis la mélange. À vous de 'reconstruire' l'image initiale.

Pour conclure, notons que le modèle original demeure affiché en permanence pour comparaison, et qu'il est possible d'obtenir une nouvelle répartition en employant l'article 'mélange' du menu 'Jeux'.

Fichier 'Puzzle.Job'

```
ASM     InitManagers   Exec   Edit
ASM     SetUpMenu      Exec   Edit
ASM     SetUpWindow    Exec   Edit
ASM     CrtTabO        Exec   Edit
ASM     DrawPuzzle     Exec   Edit
ASM     Melange        Exec   Edit
ASM     Deplacement    Exec   Edit
ASM     Puzzle.asm     Exec   Edit
LINK    Puzzle.Link    Exec   Edit
RMAKER  puzzle.r       Finder Edit
```

Fichier 'Puzzle.Link'

```
[

Puzzle
Initmanagers
SetUpMenu
SetUpWindow
CrtTabO
DrawPuzzle
Melange
Deplacement
/Output Puzzle.code
/TYPE 'TEMP'

$
```

Fichier 'Puzzle.R'

```
Puzzle
APPLSAMP

* Dialogues CRTTABO
TYPE DLOG
 ,128
Conversion StartupScreen.
50 12080390
Visible NoGoAway
1
0
129

TYPEDITL
 ,129
1

StaticText
51 0 25260
Choisissez votre Puzzle Taquin .....

TYPE DLOG
 ,130
Conversion StartupScreen.
100100190400
Visible NoGoAway
1
0
131

TYPE DITL
 ,131
4

Button
60208080
CANCEL

Button
6012080180
OK

StaticText
15 20 36 300
Attention : si startupScreen

StaticText
352056300
existe il sera Modifié !!!

TYPEALRT
 ,132
100 50 200 450
133
7FFF

TYPE DITL
 ,133
2

StaticText
157035450
Creation Startupscreen Impossible !!!

Button
7520 9580
CANCEL

*menus
TypeMENU
 ,1
\14

 ,2
File
 Quitte

 ,3
Jeux
 Autrejeu
 Mélange

*Windows
Type WIND
 ,151
PUZZLE
4010280250
Visible NoGoAway
0
0

Type WIND
 ,152
MODELE
40 260280 500
Visible NoGoAway
0
0

INCLUDE     Puzzle.Code
```

Fichier 'InitManagers.Asm'

```
; InitManagers

XDEF        XInitManagers

   .TRAP    _InitGraf       $A86E
   .TRAP    _InitFonts      $A8FE
   .TRAP    _InitWindows    $A912
   .TRAP    _InitMenus      $A930
   .TRAP    _InitDialogs    $A97B
   .TRAP    _InitCursor     $A850
   .TRAP    _FlushEvents    $A032
   .TRAP    _TEInit         $A9CC

; InitManagers
XInitManagers
   PEA      -4(A5)
   _InitGraf
   _InitFonts
   MOVE.L   #$0000FFFF,D0
   _FlushEvents
   _InitWindows
   _InitMenus
   CLR.L    -(SP)
   _InitDialogs
   _TEInit
   _InitCursor

   RTS
   END
```

Fichier 'SetUpMenu.Asm'

```
; SetUpMenu

XDEF        XSetUpMenu
XREF        AppleH
XREF        FileH
XREF        JeuxH

   .TRAP    _debugger       $A9FF
   .TRAP    _GetRMenu       $A9BF
   .TRAP    _InsertMenu     $A935
   .TRAP    _DrawMenuBar    $A937
   .TRAP    _AddResMenu     $A94D

XSetUpMenu
```

```
;Menu Apple
   CLR.L      -(SP)
   MOVE       #1,-(SP)
   _GetRMenu
   MOVE.L     (SP),AppleH(A5)
   MOVE.L     (SP),-(SP)

   CLR        -(SP)
   _InsertMenu

; Add Desk Accessoiries
   MOVE.L     #'DRVR',-(SP)
   _AddResMenu

; File Menu
   CLR.L      -(SP)
   MOVE       #2,-(SP)
   _GetRMenu
   MOVE.L     (SP),FileH(A5)
   CLR        -(SP)
   _InsertMenu

;JeuxMenu
   CLR.L      -(SP)
   MOVE       #3,-(SP)
   _GetRMenu
   MOVE.L     (SP),JeuxH(A5)
   CLR        -(SP)
   _InsertMenu

   _DrawMenuBar

   RTS

   END
```

Fichier 'SetUpWindow.Asm'

```
; SetUpWindow

XDEF          XSetUpWindow

XREF          WindStore1
XREF          WindStore2
XREF          WWindow1
XREF          WWindow2

   .TRAP      _debugger        $A9FF
   .TRAP      _GetNewWindow    $A9BD
   .TRAP      _SetPort         $A873
   .TRAP      _MoveTo          $A893
   .TRAP      _DrawString      $A884

XSetUpWindow

   CLR.L      -(SP)
   MOVE       #152,-(SP)
   PEA        WindStore2(A5)
   MOVE.L     #0,-(SP)
   _GetNewWindow
   MOVE.L     (SP)+,WWindow2(A5)
   CLR.L      -(SP)
   MOVE       #151,-(SP)
   PEA        WindStore1(A5)
   MOVE.L     #-1,-(SP)
   _GetNewWindow
   MOVE.L     (SP)+,WWindow1(A5)

   MOVE.L     WWindow1(A5),-(SP)
   _SetPort

   MOVE       #20,-(SP)
   MOVE       #20,-(SP)
   _MoveTo
   PEA        'Choisir un nouveau Jeu'
   _DrawString
   MOVE       #20,-(SP)
   MOVE       #35,-(SP)
   _MoveTo
   PEA        'dans le Menu Jeux ....'
   _DrawString
   RTS

   END
```

Fichier 'CrtTabO.Asm'

```
; CrtTabO

XDEF          XcrtTabO

   .TRAP      _debugger        $A9FF
   .TRAP      _Open            $A000
   .TRAP      _Close           $A001
   .TRAP      _Read            $A002
   .TRAP      _Write           $A003
   .TRAP      _Create          $A008
   .TRAP      _Delete          $A009
   .TRAP      _Allocate        $A010
; I/O System

; Modes pour ioPosMode

fsAtMark      EQU              0
fsFromStart   EQU              1
fsFromLEOF    EQU              2
fsFromMark    EQU              3

; Permissions pour ioPermssn

fsCurPerm     EQU              0
fsRdPerm      EQU              1
fsWrPerm      EQU              2
fsRdWrPerm    EQU              3

; I/O record

ioQElSize     EQU              $32

ioCompletion  EQU              $C
ioResult      EQU              $10
ioFileName    EQU              $12
ioVRefNum     EQU              $16
ioDrvNum      EQU              $16
ioRefNum      EQU              $18

ioFileType    EQU              $1A

; zones pour _Read _Write

ioBuffer      EQU              $20
ioByteCount   EQU              $24
ioNumDone     EQU              $28

ioPosMode     EQU              $2C
ioPosOffset   EQU              $2E

; zones pour _Allocate

ioReqCount    EQU              $24
ioActCount    EQU              $28
; zones pour _Open

ioPermssn     EQU              $1B
ioOwnBuf      EQU              $1C

   .TRAP      _Pack3           $A9EA

; Reply record data structure
rGood         EQU              0
rType         EQU              2
rVolume       EQU              6
rVersion      EQU              8
rName         EQU              10

sfGetFile     EQU              2

; Dialogues
   .TRAP      _GetNewDialog    $A97C
   .TRAP      _Alert           $A985
   .TRAP      _SetPort         $A873
   .TRAP      _ModalDialog     $A991
   .TRAP      _CloseDialog     $A982
   .TRAP      _StopAlert       $A986

dWindLen      EQU              $AA

; Macro Lect
MACRO         _Lect R1 =
   LEA        Buf(A5),A6
   MOVE.L     A6,ioBuffer(A0)
   MOVE.L     A6,A1

   MOVE.L     #511,D5
{R1}
   CLR.B      (A6)+
   DBRA       D5,{R1}

   _Read

   MOVE.L     A1,A2
   ADD.L      ioNumDone(A0),A2
   |

; Macro Rout
MACRO         _Rout R1,R2,R3,R4 =
   ADDQ       #1,D4
   CMP        #(240/8),D4
   BLE        {R1}
   CMP        #72,D4
   BNE        {R2}
   CLR        D4
{R2}
   TST.B      -(A3)
{R1}
   CMPA.L     A3,A4
   BEQ        FinRout
   CMPA.L     A1,A2
   BNE        {R3}
   CMP        #-39,ioResult(A0)
   BEQ        FinRout
   _Lect      {R4}
{R3}
   |

XCrtTabO
; Dialog explications
   CLR.L      -(SP)
   MOVE       #128,-(SP)
   PEA        DStorage
   MOVE.L     #-1,-(SP)
   _GetNewDialog

   MOVE.L     (SP)+,A6

; Standard File
   MOVE       #82,-(SP)
   MOVE       #124,-(SP)
   CLR.L      -(SP)
   CLR.L      -(SP)
   MOVE       #1,-(SP)
   PEA        sfTypeList
   CLR.L      -(SP)
   PEA        sfReply(A5)
   MOVE.W     #sfGetFile,-(SP)
   _Pack3

   MOVE.L     A6,-(SP)
   _CloseDialog

   LEA        sfReply(A5),A0
   MOVE       rGood(A0),D4
```

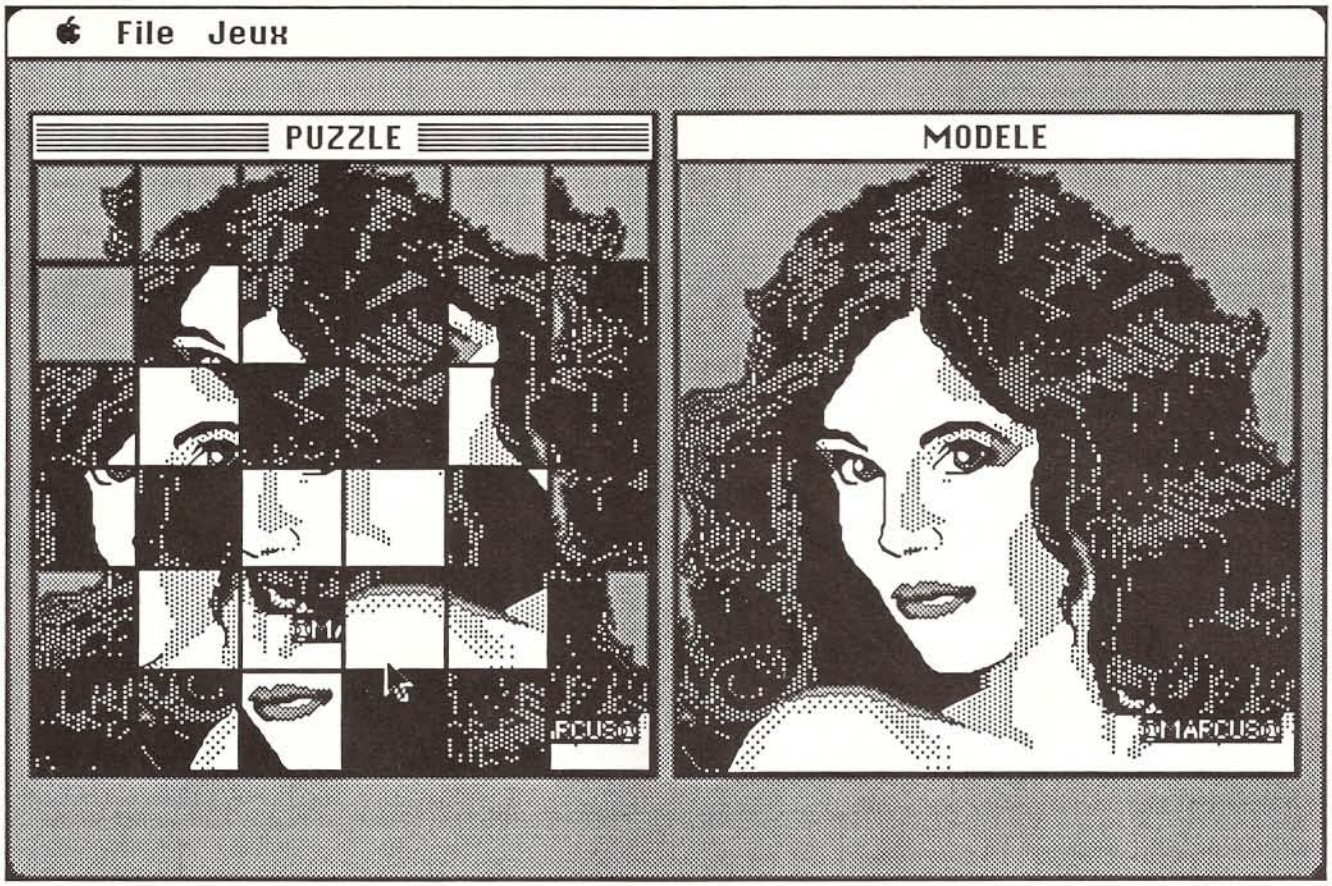
```
    BNE         OpenPaint
    RTS
; Open fichier MacPaint
OpenPaint
    MOVE.L      A0,A1
    LEA         DiskI(A5),A0
    CLR.L       ioCompletion(A0)
    LEA         rName(A1),A6
    MOVE.L      A6,ioFileName(A0)
    MOVE        rVolume(A1),
                ioVRefNum(A0)
    CLR.B       ioFileType(A0)
    MOVE.B      #fsRdPerm,ioPermssn(A0)
    CLR.L       ioOwnBuf(A0)
    _Open

; Rab tableau sortie
RabTabO
    CLR.L       D2
    MOVE        #((240/8)*240),D2
    SUBQ.L      #1,D2
    MOVE.L      A3,A6
RabO
    CLR.B       (A6)+
    DBRA        D2,Rabo

; Routine conversion
    LEA         DiskI(A5),A0
    MOVE.L      #512,ioByteCount(A0)
    MOVE        #fsFromStart,ioPosMode(A0)
    MOVE.L      #512,ioPosOffset(A0)

    MOVE.L      A3,A4
    ADD.L       #((240/8)*240),A4

    CLR         D4

    _Lect       K0

Rout
    CLR.L       D2
    MOVE.B      (A1)+,D2
    CMPA.L      A1,A2
    BNE         TestB
    CMP         #-39,ioResult(A0)
    BEQ         FinRout
    _Lect       I0
TestB
    BTST        #7,D2
    BNE         Bit7a1d
Bit7a0
    MOVE.B      (A1)+,(A3)+
    _Rout       E0,F0,G0,H0
    DBRA        D2,Bit7a0
    BRA         Rout
Bit7a1d
    NEG.L       D2
    ADD.L       #256,D2
Bit7a1
    MOVE.B      (A1),(A3)+
    _Rout       E1,F1,G1,H1
    DBRA        D2,Bit7a1
    TST.B       (A1)+
    CMPA.L      A1,A2
    BNE         Rout
    CMP         #-39,ioResult(A0)
    BEQ         FinRout
    _Lect       J0
    BRA         Rout
FinRout

; Close du Fichier
    _Close
    RTS
; Data
DStorage        DCB.W       DWindLen,0
ItemHit         DC          0

; Global Data
DiskI           DS.B        ioQElSize
Buf             DS.B        512

sfReply         DS.B        72
sfTypeList      DC.B        'PNTG'

    END
```

Fichier 'DrawPuzzle.Asm'

```
; DrawPuzzle

XDEF            XDrawPuzzle

XREF            WWindow1
XREF            WWindow2

    .TRAP       _Debugger       $A9FF
    .TRAP       _PenPat         $A89D
    .TRAP       _PenSize        $A89B
    .TRAP       _PenNormal      $A89E
    .TRAP       _MoveTo         $A893
    .TRAP       _LineTo         $A891
    .TRAP       _FrameRect      $A8A1
    .TRAP       _CopyBits       $A8EC
    .TRAP       _TextMode       $A889
    .TRAP       _TextSize       $A88A
    .TRAP       _DrawString     $A884
    .TRAP       _SetPort        $A873
    .TRAP       _MoveWindow     $A91B
    .TRAP       _PaintRect      $A8A2

XDrawPuzzle

    MOVE.L      WWindow1(A5),-(SP)
    MOVE        #10,-(SP)
    MOVE        #40,-(SP)
    CLR         -(SP)
    _MoveWindow
    MOVE.L      WWindow2(A5),-(SP)
    MOVE        #260,-(SP)
    MOVE        #40,-(SP)
    CLR         -(SP)
    _MoveWindow

    MOVE.L      WWindow1(A5),-(SP)
    _SetPort

    LEA         Bitmap,A6
    MOVE.L      A3,(A6)
    MOVE.L      A6,-(SP)
    MOVE.L      WWindow1(A5),A6
    ADDA.L      #2,A6
    MOVE.L      A6,-(SP)
    PEA         RectBitMap
    PEA         RectBitMap
    CLR         -(SP)
    CLR.L       -(SP)
    _CopyBits

    MOVE        #9,-(SP)
    _TextSize
    MOVE        #4,-(SP)
    _TextMode
    MOVE        #180,-(SP)
    MOVE        #227,-(SP)
    _MoveTo
    PEA         '@MARCUS@'
    _DrawString

    MOVE.L      WWindow1(A5),A6
    ADDA.L      #2,A6
    MOVE.L      A6,-(SP)
    MOVE.L      WWindow2(A5),A6
    ADDA.L      #2,A6
    MOVE.L      A6,-(SP)
    PEA         ViewRect
    PEA         ViewRect
    MOVE        #0,-(SP)
    CLR.L       -(SP)
    _CopyBits

    MOVE.L      WWindow1(A5),-(SP)
    _SetPort

    _PenNormal

    PEA         ViewRect
    _FrameRect

    MOVE        #2,-(SP)
    MOVE        #2,-(SP)
    _PenSize

    MOVE        #39,D2
    CLR         D3
    MOVE        #39,D4
    MOVE        #240,D5

    MOVEQ       #5,D6

GrilleV
    MOVE        D2,-(SP)
    MOVE        D3,-(SP)
    _MoveTo
    MOVE        D4,-(SP)
    MOVE        D5,-(SP)
    _LineTo
    ADD         #40,D2
    ADD         #40,D4
    DBRA        D6,GrilleV

    CLR         D2
    MOVE        #39,D3
    MOVE        #240,D4
    MOVE        #39,D5

    MOVEQ       #5,D6

GrilleH
    MOVE        D2,-(SP)
    MOVE        D3,-(SP)
    _MoveTo
    MOVE        D4,-(SP)
    MOVE        D5,-(SP)
    _LineTo
    ADD         #40,D3
    ADD         #40,D5
    DBRA        D6,GrilleH

    MOVE        #1,-(SP)
    MOVE        #1,-(SP)
    _PenSize

    PEA         RectNoir
    _PaintRect

    CLR.L       D7

    RTS

ViewRect        DC          0,0,240,240
RectNoir        DC          0,0,39,39
BitMap          DC.L        0
OctLign         DC          30
RectBitMap      DC          0,0,240,240

Pat             DS.B        8
```

Fichier 'Melange.Asm'

```
; Melange

XDEF            XMelange
XREF            WWindow1
```




```
   .TRAP    _Debugger      $A9FF
   .TRAP    _CopyBits      $A8EC
   .TRAP    _PaintRect     $A8A2
   .TRAP    _Random        $A861
   .TRAP    _SetPort       $A873

XMelange
   MOVE.L   WWindow1(A5),-(SP)
   _SetPort

   CMP      #99,D7
   BEQ      FinM

   CLR      D5
   CLR.L    D3

Hasard
   CLR      -(SP)
   _Random
   MOVE     (SP)+,D3

   CMP      #-16384,D3
   BGT      @1
   MOVE     #(1*8),D3
   BRA      MoveCarre
@1
   CMP      #0,D3
   BGT      @2
   MOVE     #-(1*8),D3
   BRA      MoveCarre
@2
   CMP      #+16384,D3
   BGT      @3
   MOVE     #(6*8),D3
   BRA      MoveCarre
@3
   MOVE     #-(6*8),D3

MoveCarre

   MOVE.L   D7,D6
   ADD      D3,D6
   CMP      #0,D6
   BLT      Hasard
   CMP      #35*8,D6
   BGT      Hasard

   MOVE.L   WWindow1(A5),A6
   ADDA.L   #2,A6
   MOVE.L   A6,-(SP)
   MOVE.L   A6,-(SP)
   LEA      PosRectM,A3
   ADDA.L   D6,A3
   MOVE.L   A3,-(SP)
   LEA      PosRectM,A4
   ADDA.L   D7,A4
   MOVE.L   A4,-(SP)
   MOVE     #0,-(SP)
   CLR.L    -(SP)
   _CopyBits

   MOVE.L   A3,-(SP)
   _PaintRect

   MOVE.L   D6,D7

   ADDQ     #1,D5
   CMPI     #160,D5
   BNE      Hasard

FinM
   RTS

PosRectM

   DC       000,000,039,039,000,040
   DC       039,079,000,080,039,119
   DC       000,120,039,159,000,160
   DC       039,199,000,200,039,239
   DC       040,000,079,039,040,040
   DC       079,079,040,080,079,119
   DC       040,120,079,159,040,160
   DC       079,199,040,200,079,239
   DC       080,000,119,039,080,040
   DC       119,079,080,080,119,119
   DC       080,120,119,159,080,160
   DC       119,199,080,200,119,239
   DC       120,000,159,039,120,040
   DC       159,079,120,080,159,119
   DC       120,120,159,159,120,160
   DC       159,199,120,200,159,239
   DC       160,000,199,039,160,040
   DC       199,079,160,080,199,119
   DC       160,120,199,159,160,160
   DC       199,199,160,200,199,239
   DC       200,000,239,039,200,040
   DC       239,079,200,080,239,119
   DC       200,120,239,159,200,160
   DC       239,199,200,200,239,239

   END
```

Fichier 'Deplacement.Asm'

```
; Deplacement

XDEF        XDeplacement
XREF        WWindow1

   .TRAP    _Debugger      $A9FF
   .TRAP    _PtInRect      $A8AD
   .TRAP    _CopyBits      $A8EC
   .TRAP    _PaintRect     $A8A2
   .TRAP    _GetMouse      $A972

XDeplacement

   CMP      #99,D7
   BEQ      FinD
   PEA      XYMouse(A5)
   _GetMouse
```

```
    CLR.L     D2
AutreCarre
    CLR       -(SP)
    MOVE      XYMouse+2(A5),-(SP)
    MOVE      XYMouse(A5),-(SP)
    LEA       PosRect,A4
    ADDA.L    D2,A4
    MOVE.L    A4,-(SP)
    _PtInRect
    MOVE      (SP)+,D3
    BNE       Trouve
    CMP.L     #(8*35),D2
    BEQ       FinD
    ADDQ.L    #8,D2
    BRA       AutreCarre


Trouve
    CLR.L     D3
    MOVE      D2,D3
    ADD       #-(1*8),D3
    CMP       D7,D3
    BEQ       Contigu
    MOVE      D2,D3
    ADD       #(1*8),D3
    CMP       D7,D3
    BEQ       Contigu
    MOVE      D2,D3
    ADD       #-(6*8),D3
    CMP       D7,D3
    BEQ       Contigu
    MOVE      D2,D3
    ADD       #(6*8),D3
    CMP       D7,D3
    BEQ       Contigu
    BRA       FinD

Contigu
    MOVE.L    WWindow1(A5),A6
    ADDA.L    #2,A6
    MOVE.L    A6,-(SP)
    MOVE.L    A6,-(SP)
    MOVE.L    A4,-(SP)
    LEA       PosRect,A6
    ADDA.L    D7,A6
    MOVE.L    A6,-(SP)
    MOVE      #0,-(SP)
    CLR.L     -(SP)
    _CopyBits
    MOVE.L    A4,-(SP)
    _PaintRect

    MOVE      D2,D7

FinD
    RTS

PosRect

    DC        000,000,039,039,000,040
    DC        039,079,000,080,039,119
    DC        000,120,039,159,000,160
    DC        039,199,000,200,039,239
    DC        040,000,079,039,040,040
    DC        079,079,040,080,079,119
    DC        040,120,079,159,040,160
    DC        079,199,040,200,079,239
    DC        080,000,119,039,080,040
    DC        119,079,080,080,119,119
    DC        080,120,119,159,080,160
    DC        119,199,080,200,119,239
    DC        120,000,159,039,120,040
    DC        159,079,120,080,159,119
    DC        120,120,159,159,120,160
    DC        159,199,120,200,159,239
    DC        160,000,199,039,160,040
    DC        199,079,160,080,199,119
    DC        160,120,199,159,160,160
    DC        199,199,160,200,199,239
    DC        200,000,239,039,200,040
    DC        239,079,200,080,239,119
    DC        200,120,239,159,200,160
    DC        239,199,200,200,239,239

XYMouse     DS        2

    END
```

Fichier 'Puzzle.Asm'

```
XREF        XInitManagers
XREF        XSetUpMenu
XREF        XSetUpWindow
XREF        XDeplacement
XREF        XMelange
XREF        XDrawPuzzle
XREF        XCrtTabO

XDEF        WindStore1
XDEF        WindStore2
XDEF        WWindow1
XDEF        WWindow2
XDEF        AppleH
XDEF        FileH
XDEF        JeuxH

    .TRAP   _debugger       $A9FF

    .TRAP   _GetNewDialog   $A97C
    .TRAP   _Alert          $A985
    .TRAP   _SetPort        $A873
    .TRAP   _ModalDialog    $A991
    .TRAP   _CloseDialog    $A982
    .TRAP   _StopAlert      $A986

dWindLen    EQU             $AA

    .TRAP   _NewHandle      $A122
    .TRAP   _DisposHandle   $A023
    .TRAP   _HLock          $A029
    .TRAP   _HUnLock        $A02A
    .TRAP   _GetRMenu       $A9BF
    .TRAP   _InsertMenu     $A935
    .TRAP   _DrawMenuBar    $A937
    .TRAP   _AddResMenu     $A94D
    .TRAP   _MenuSelect     $A93D
    .TRAP   _HiliteMenu     $A938

EvtBlkSize  EQU             $10

evtNum      EQU             0
evtMessage  EQU             2
evtTicks    EQU             6
evtMouse    EQU             $A
evtMeta     EQU             $E
evtMBut     EQU             $F

    .TRAP   _FindWindow     $A92C
    .TRAP   _DragWindow     $A925
    .TRAP   _SelectWindow   $A91F
    .TRAP   _SystemClick    $A9B3
    .TRAP   _GetItem        $A946
    .TRAP   _OpenDeskAcc    $A9B6
    .TRAP   _BeginUpdate    $A922
    .TRAP   _EndUpdate      $A923
    .TRAP   _CopyBits       $A8EC
    .TRAP   _EraseREct      $A8A3
    .TRAP   _SystemTask     $A9B4
    .TRAP   _GetNextEvent   $A970

WindowSize  EQU             $9C

    BSR     XInitManagers
    BSR     XSetUpMenu
    BSR     InitBitMap
    CLR.B   Tag(A5)
    BSR     XSetUpWindow
    MOVE.L  #99,D7

EventLoop
    _SystemTask

    CLR     -(SP)
    MOVE    #$0FFF,-(SP)
    PEA     EventRecord
    _GetNextEvent

    MOVE    (SP)+,D0
    BEQ     EventLoop
    BSR     HandleEvent
    BEQ     EventLoop

    RTS

HandleEvent
    MOVE    EventRecord+evtNum(A5),
            D0
    ADD     D0,D0
    MOVE    EventTable(D0),D0
    JMP     EventTable(D0)

EventTable
    DC      NextEvent-EventTable
    DC      MouseDown-EventTable
    DC      NextEvent-EventTable
    DC      NextEvent-EventTable
    DC      NextEvent-EventTable
    DC      NextEvent-EventTable
    DC      Update-EventTable
    DC      NextEvent-EventTable
    DC      Activate-EventTable
    DC      NextEvent-EventTable
    DC      NextEvent-EventTable
    DC      NextEvent-EventTable

Activate
    MOVE.L  EventRecord+
            evtMessage(A5),D2
    MOVE.L  D2,WWindow(A5)
    CMP.L   WWindow1(A5),D2
    BEQ     @1
    CMP.L   WWindow2(A5),D2
    BNE     NextEvent
@1
    MOVE    EventRecord+
            evtMeta(A5),D3
    BTST    #0,D3
    BNE     NextEvent
    MOVE.L  D2,-(SP)
    _SetPort
@2
    BRA     NextEvent

Update
    MOVE.L  EventRecord+
            evtMessage(A5),
            WWindow(A5)
    MOVE.L  WWindow(A5),-(SP)
    _BeginUpDate

    TST.B   Tag(A5)
    BEQ     EndMaj
    MOVE.L  WWindow(A5),D2
    CMP.L   WWindow1(A5),D2
    BNE     @11
    BSR     RstPuzzle
@11
    CMP.L   WWindow2(A5),D2
    BNE     EndMaj
    BSR     RstModele

EndMaj
    MOVE.L  WWindow(A5),-(SP)
    _EndUpdate
    BRA     NextEvent

MouseDown
```

```
    CLR         -(SP)
    MOVE.L      EventRecord+
                EvtMouse(AS),-(SP)
    PEA         WWindow(A5)
    _FindWindow

    MOVE        (SP)+,D0
    ADD         D0,D0
    MOVE        WindowTable(D0),D0
    JMP         WindowTable(D0)

WindowTable
    DC.W        NextEvent-WindowTable
    DC.W        InMenu-WindowTable
    DC.W        SystemEvent-WindowTable
    DC.W        Content-WindowTable
    DC.W        Drag-WindowTable
    DC.W        NextEvent-WindowTable
    DC.W        NextEvent-WindowTable

InMenu
    CLR.L       -(SP)
    MOVE.L      EventRecord+
                evtMouse(A5),-(SP)
    _MenuSelect

    MOVE        (SP)+,Menu(A5)
    MOVE        (SP)+,MenuItem(A5)

    MOVE        Menu(A5),D2
    CMP         #1,D2
    BEQ         InAppleMenu
    CMP         #2,D2
    BEQ         InFileMenu
    CMP         #3,D2
    BEQ         InJeuxMenu

ChoixReturn
    CLR         -(SP)
    _HiliteMenu
    BRA         NextEvent

InAppleMenu
    MOVE.L      AppleH(A5),-(SP)
    MOVE        MenuItem(A5),-(SP)
    PEA         DeskName(A5)
    _GetItem

    CLR         -(SP)
    PEA         DeskName(A5)
    _OpenDeskAcc
    MOVE        (SP)+,D0

    MOVE.L      WWindow1(A5),-(SP)
    _SetPort

    BRA         ChoixReturn
InFileMenu
    MOVE        #$FF,D0
    RTS
InJeuxMenu
    CMP         #1,MenuItem(A5)

    BNE         Choix32
    BSR         NewJeu
    BRA         ChoixReturn
Choix32
    CMP         #2,MenuItem(A5)
    BNE         ChoixReturn
    BSR         XMelange
    BSR         SavePuzzle
    BRA         ChoixReturn

SystemEvent
    PEA         EventRecord(A5)
    MOVE.L      WWindow(A5),-(SP)
    _SystemClick
    BRA         NextEvent
Content
    MOVE.L      WWindow(A5),D2
    CMP.L       WWindow1(A5),D2
    BNE         NextEvent
    MOVE.L      D2,-(SP)
    _SetPort
    MOVE.L      D2,-(SP)
    _SelectWindow
    BSR         RstPuzzle
    BSR         XDeplacement
    BSR         SavePuzzle
    BRA         NextEvent
Drag
    MOVE.L      WWindow(A5),-(SP)
    MOVE.L      EventRecord+
                evtMouse(A5),-(SP)
    PEA         Bounds
    _DragWindow

    BRA         NextEvent

NextEvent
    MOVE        #$00,D0

    RTS

NewJeu
    MOVE.L      #((240/8)*240),D0
    _NewHandle
    BNE         @1
    _HLock

    MOVE.L      A0,Handle(A5)
    MOVE.L      (A0),A3
    BSR         XCrtTabO
    TST         D4
    BEQ         @2

    MOVE.L      Handle(A5),A0
    MOVE.L      (A0),A3
    BSR         XDrawPuzzle

    BSR         Xmelange
    BSR         SavePuzzle
    BSR         SaveModele

@2
    MOVE.L      Handle(A5),A0
    _HUnLock
    _DisposHandle

    MOVE.B      #$FF,Tag(A5)

@1
    RTS

InitBitMap
    LEA         TMap1(A5),A6
    MOVE.L      A6,Ptr1(A5)
    MOVE        #30,Row1(A5)
    CLR.L       Bounds1+0(A5)
    MOVE        #240,Bounds1+4(A5)
    MOVE        #240,Bounds1+6(A5)
    LEA         TMap2(A5),A6
    MOVE.L      A6,Ptr2(A5)
    MOVE        #30,Row2(A5)
    CLR.L       Bounds2+0(A5)
    MOVE        #240,Bounds2+4(A5)
    MOVE        #240,Bounds2+6(A5)
    RTS

SavePuzzle
    MOVE.L      WWindow1(A5),A6
    ADDA.L      #2,A6
    MOVE.L      A6,-(SP)
    PEA         Ptr1(A5)
    PEA         ViewRect
    PEA         ViewRect
    MOVE        #0,-(SP)
    CLR.L       -(SP)
    _CopyBits
    RTS

RstPuzzle
    PEA         Ptr1(A5)
    MOVE.L      WWindow1(A5),A6
    ADDA.L      #2,A6
    MOVE.L      A6,-(SP)
    PEA         ViewRect
    PEA         ViewRect
    MOVE        #0,-(SP)
    CLR.L       -(SP)
    _CopyBits
    RTS

SaveModele
    MOVE.L      WWindow2(A5),A6
    ADDA.L      #2,A6
    MOVE.L      A6,-(SP)
    PEA         Ptr2(A5)
    PEA         ViewRect
    PEA         ViewRect
    MOVE        #0,-(SP)
    CLR.L       -(SP)
    _CopyBits
    RTS

RstModele
    PEA         Ptr2(A5)
    MOVE.L      WWindow2(A5),A6
    ADDA.L      #2,A6
    MOVE.L      A6,-(SP)
    PEA         ViewRect
    PEA         ViewRect
    MOVE        #0,-(SP)
    CLR.L       -(SP)
    _CopyBits
    RTS

; Data
DStorage        DCB.W   DWindLen,0
Item1Hit        DC      0
Bounds          DC      28,4,308,508
ViewRect        DC      0,0,240,240

; Global Data

DeskName        DS      16

WWindow         DS.L    1
WWindow1        DS.L    1
WWindow2        DS.L    1

WindStore1      DS.B    WindowSize
WindStore2      DS.B    WindowSize

EventRecord     DS.B    evtBlkSize

AppleH          DS.L    1
FileH           DS.L    1
JeuxH           DS.L    1
Menu            DS.L    1
MenuItem        DS.L    1

Handle          DS.L    1

Ptr1            DS.L    1
Row1            DS      1
Bounds1         DS      4
Tmap1           DS.B    7200

Ptr2            DS.L    1
Row2            DS      1

Bounds2         DS      4
Tmap2           DS.B    7200

Tag             DS.B    1

    END
```

Macintosh : les news

Ariel Sebban

RIEN ! Il ne s'est rien passé, 40 000 visiteurs, 400 exposants, près de 200 000 m2 d'exposition, il ne s'est rien passé à Boston ! Hé oh! Vous trouvez pas que certains exagèrent ? Que leur faut-il ? Un nouveau Mac tous les deux jours ? Ou peut-être des effets d'annonce toutes les deux heures ?

Non soyons sérieux, le monde Macintosh est désormais complètement majeur, et si Apple ne sort rien de fabuleux pour Apple Expo, les nouveautés logicielles et matérielles foisonnent.

Nouveautés logicielles

C'est la guerre tous azimuts, traitements de texte, tableurs, base de données, logiciels de présentation en tout genre, comment dites vous "desktop quelque chose", DAO, CAO, etc. tout y passe et l'on frise l'apoplexie. Mais tous nécessitent plus de mémoire vive, plus de mémoire de masse, plus de vitesse, et les marchands de périphériques se frottent les mains. *Qui n'a pas ses 4 mégas de MEV , un petit disque de 150 mégas pour monsieur ? Comment ? Mais sans 68030 à 20 Mhz point de salut.*

Pourtant, il vous la faudra cette extension (malgré la crise) pour faire tourner pleinement le nouveau système baptisé 6.0x.

Ce système corrige quelques bugs du MultiFinder 1.0 et arrive avec des utilitaires comme un (petit) générateur de macros, une fonction de loupe sur le travail en cours et une horloge universelle. On remarque quelques incompatibilités logicielles avec le nouveau système et en particulier avec le générateur de macros. Sont dors et déjà complètement compatibles : Microsoft Word 3.02, 4ème Dimension, Turbo Pascal 1.1, MacWrite 5.0, MacPaint 2.0, Mac Draw II, SuperPaint 1.1 et Mac Project II. MS Works 1.1 présente des problèmes avec le générateur de macros, et les versions antérieures à 1.06 d'Excel ont une certaine incompatibilité d'humeur. Microsoft recommande de passer à la version 1.5. Prudence donc avant de jeter votre ancien système à la poubelle.

Traitements de textes

Fini le temps des bidouilleurs. Le Mac est un outil *professionnel Monsieur. Comment, mais vous utilisez MacWrite, mais quelle horreur.* Mais aujourd'hui tout traitement de texte qui se respecte doit être capable de produire l'édition de fin de semaine de votre quotidien avec ses suppléments en couleurs *Monsieur !*

En tout cas c'est ce qu'annoncent à peu de chose près **FullWrite Professionnal** d'**Ashton Tate**, et **WordPerfect**. Deux monstres du traitement de texte, qui vous permettrons de sortir vos lettres pour belle maman avec une qualité incomparable. Qu'est ce qui vous manque dans votre traitement de textes actuel ? Allez dites, c'est pas la peine de réfléchir, maintenant ça existe. De la simple saisie à la mise en page la plus sophistiquée, avec multicolonnage, détourage d'images de toute provenance, tout y est. Le seul problème c'est qu'on est en train d'oublier la philosophie première du Mac : la simplicité ; et comme corollaire une utilisation quasi intuitive. Les deux se perdent et il faut désormais se perdre dans les labyrinthes de documentations de plus en plus épaisses et souvent obscures. Tout compte fait **MacWrite** de **Claris** maintenant en version 5.0 suffit souvent largement.

Tableurs

Les tableurs ne sont pas en reste. **Excel** de **Microsoft** dans sa version 1.5 prend de la couleur, gagne quelques 44 fonctions et des menus personnalisables. **Wingz** d'**Informix** devrait prendre la relève pour les années 90, mais n'en n'a pas fini d'être mis au point, et **Full Impact** vient compléter l'offre Macintosh pour **Ashton Tate**.

Bases de données

4ème dimension d'**ACI**, écrasant tous ses concurrents, est-il encore utile de citer **FoxBase+** de **Fox Software**, capable de faire tourner des application de dBase III+ du monde MS/DOS (beurk) et ce sans modifications, mais qui s'en soucie vraiment... Passons.

Hypercard

Une version 1.2 est annoncée avec de nouvelles fonctionnalités — le rendant plus puissant — qui le fait se rapprocher de plus en plus d'une base de données hors du commun. Dans le domaine des utilitaires, **Reports** d'**Activision** permet de combler les lacunes d'Hypercard en matière de rapports.

Il semble qu'en France, Hypercard n'ait pas encore rencontré l'enthousiasme qu'on lui connaît outre Atlantique. Car la grosse question reste ici : que faire avec ? Et pendant qu'on essaie de trouver une réponse en France, la chaîne ABC aux U.S.A. utilise 130 piles et 5 000 cartes pour sa gestion des élections présidentielles. Personnellement je m'en sert pour gérer la copropriété dans laquelle j'habite, et aucun autre langage m'aurait permis de faire la même chose aussi simplement. Et pourquoi pas une

rubrique HyperCard dans Pom's. Vos contributions seraient les bienvenues.

Accessoires de bureau

Ils n'ont plus d'accessoires que le nom, et nombreux sont ceux qui se révèlent rapidement indispensables.

DiskToolsII d'Electronic Arts met un Finder en menu , **Quick Keys** de **CE software** permet d'utiliser pleinement les possibilités des claviers étendus par la création de macros assignées à une seule touche, **Suit Case** de **Fifth Generation System's** manipule polices de caractères et accessoires de bureau sans passer par Font/DA Mover. **SuperLaser Spool 2.0** de **SuperMac**, désormais compatible MultiFinder, reste le meilleur spooler à ce jour. Nous tâcherons de consacrer une plus grande partie de cette rubrique prochainement à la constitution du menu idéal.

Utilitaires divers

Disk Fit 1.2 de **SuperMac** et **Redux 1.0** de **Microseeds**, vous permettrons des sauvegardes rapides et efficaces de vos disques durs. **Disk Express 1.5** de **ALSoft, Inc** vous permettra d'optimiser votre espace disque.

Et dans la rubrique "il faut-en parler" : les virus. Sans attribuer à ces sales bestioles tous les "Boum! une grave erreur système ID xx...", il est souhaitable de prendre des précautions élémentaires. En particulier, il existe un ensemble d'utilitaires (**Virus RX**, **Virus détective**, etc.) dont certains signés **Apple**, qu'il est indispensable d'avoir dans son dossier système. Et, croyez moi, la première des choses sur laquelle mon disque dur démarre c'est **Vaccine INIT** de **CE Software**. Alors prudence, et testez toujours toute nouveauté sur disquette verrouillée, disque dur déconnecté.

Les titres cités ne couvrent malheureusement qu'une infime partie de toutes les nouveautés logicielles, et ce numéro entier de Pom's n'y suffirait pas, mais il existe une bible en la matière qui est **MacGuide** qui recense environ 3 500 produits en tout genre, leur prix, les adresses, et un rapide descriptif de leurs possibilités.

Matériel

Durs les disques !

Question : pourquoi en France les disques durs pour Mac sont-ils toujours aussi chers ? Avec un dollar à 6,50 F, rien ne justifie de tels écarts de prix. Les fabricants Français ne sont-ils donc pas capable de sortir un disque dur pour moins de 4 000 F TTC. Le Mac et ses périphériques sont-ils condamnés à rester en France des produits de luxe ? Faites un effort, Messieurs. Un Dayna 60 mégas ne coûte que 720 $ aux États-Unis, soit environ 4 600 francs !

Dans la même rubrique, les cartes accélératrices qui commencent là-bas à environ 5 000 francs, coûtent ici au moins le double. Il faudra qu'on m'explique. Et on trouve des modems compatibles Hayes (Anchor Automation, Inc) à 240 $.

Les grands écrans sont désormais nombreux et vous pourrez trouver un test complet dans le numéro d'août de **MACazine**.

Dans la gamme des nouveautés matérielles, ce sont surtout les périphériques à destination graphique qui se détachent. Scanners, dont un signé Apple, interfacé HyperCard, et autres éditeurs de diapositives couleurs tel **Image Maker** de **Presentation Technologies** se multiplient. Mais encore hors de prix pour un utilisateur de base.

Nous tâcherons dans nos prochains numéros de cibler le sujet de cette rubrique. Et — Pom's oblige — nous ferons dès le prochain numéro, une revue des langages. D'hyperTalk à MPW en passant par ZBasic, il y en a pour tous les goûts, et ne croyez plus que le Mac soit si difficile à programmer. C'est juste une question de philosophie !

Adresses

ACI
5, rue Beaujon
75008 Paris
☎ (1) 42 27 37 25

Activision
3885 Bohannon Dr.
Menlo Park, CA 94025

Anchor Automation, Inc
20675 Bahama Street
Chastworth,
CA 91406

Ashton Tate - LCE
La Mare à Jouy - Douains
27120 Pacy sur Eure
☎ 32 64 63 62

CE Software
P.O Box 65580
West Des Moines, IA
50265

Claris P.Ingénierie
10, rue Mercoeur
75011 Paris
☎ (1) 43 70 71 72

Dayna
50 S Main St.
Suite 530
Salt Lake City, UT 84144

Electronic Arts
1820 Gateway Dr.
San Mateo, CA 94404

Fifth Generation System's
1322 Bell Ave.
Suite 1A
Irvine, CA 94123

Fox Software
118 W. South Bondary
Perrysburg, OH 43551

Informix
6, bd du Général Leclerc
92110 Clichy
☎ (1) 42 70 67 74

MACazine
#919, P.O Box 9802
Austin, Texas 78766

MacGuide
550 South Wadsworth blvd
Suite 500
Lakewood, CO 80226

Microseeds
4702 N. Hesperides
Tampa, FL 3361

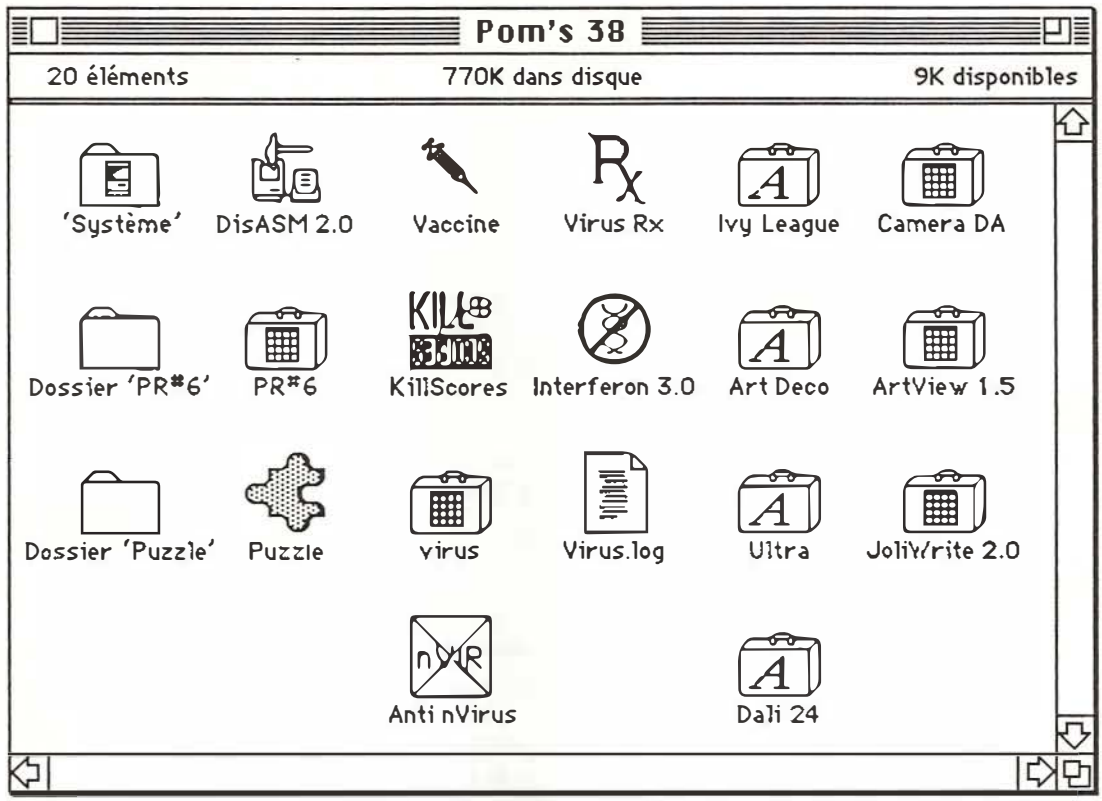
Microsoft
12, avenue du Québec
Z.A de Courtaboeuf
91957 Les Ulis Cedex

Presentation Technologies
743 N. Pastoria Ave.
Sunnyvale, CA 94086

SuperMac Symbiotic
4, rue Robert Schumann
94220 Charenton
☎ (1) 43 78 99 99

WordPerfect corp
1555 North Technology Way
Orem, UT 84057

Disquette Pom's Mac 38



Anti-Virus

La disquette Pom's Mac 38 accompagnant ce numéro contient tous les outils nécessaires pour venir à bout des virus qui — éventuellement — envahiraient vos programmes et fichiers. Ces outils se présentent sous différentes formes :

Virus Rx, **KillScores**, **Anti nVirus** et **Interferon 3.0** sont des applications à lancer avec un 'double-clic' depuis le Finder ;

Virus est un accessoire de bureau à installer dans le fichier système de votre disquette de démarrage avec Font/DA Mover ;

Vaccine est un programme qui, une fois installé dans le dossier système de votre disquette de démarrage, sera appelé depuis le tableau de bord suivant le même principe que Général, Frappe clavier, Souris, etc.

Polices de caractères

La disquette 38 contient les polices **Ivy League**, **Art Deco**, **Ultra** et **Dali**.

Accessoires de bureau

Camera DA est un accessoire de bureau qui produit le même travail que les combinaisons de touches Commande - majuscule - 3 ou Commande - majuscule - 4, c'est-à-dire la copie d'écran sur ImageWriter ou sous la forme d'un fichier de type 'Paint'. Il présente toutefois quelques avantages par rapport à la fonction classique :

- la copie d'écran est toujours possible, même avec les applications qui ne l'autorise pas ;
- l'exécution de la copie d'écran peut être différée de 1 à 60 secondes ;
- la copie avec un menu déroulé est possible (comme avec les Mac 128 et 512Ko) ;
- Camera DA permet aussi d'obtenir des copies d'écran avec curseur masqué.

ArtView est un mini 'MacPaint' toujours disponible dans le menu ☛. Écrit par Jean-Yves Thuaudet.

JoliWrite est un traitement de textes aux possibilités bien sûr plus limitées que, par exemple, MacWrite, mais qui a l'avantage d'être toujours disponible.

PR#6 est l'accessoire de redémarrage présenté dans ce numéro de Pom's.

Applications

DisASM 2.0 est un désassembleur symbolique qui raviera les explorateurs de mémoires et fichiers. Nous devons cette application à Jean-Yves Thuaudet.

Puzzle est une application ludique entièrement écrite en assembleur, et présentée dans ce numéro.

Où l'on découvre ✧ le GS le moins cher ✧ ses premiers virus ✧ une forme de vandalisme ✧ la guerre, l'amour ✧ les cerveaux de l'Apple // ✧ William Mensch ✧ les accélérateurs ✧ le //c 'speediské' ✧ le rongeur en cage ✧ le Mac caché du GS ✧ l'AppleWorks 2.1 ✧ les dumps écran ✧ le 68881 ✧ le Canada Dry des CDA ✧ le Système 3.2 ✧

Les infos : Apple // for ever

Jean-Yves Bourdin

Nouvelles du front

Vive la France !

Quel est l'ordinateur le plus vendu aux USA dans les boutiques d'informatique personnelle ? Le GS bien sûr. Les acheteurs américains sont d'ailleurs très heureux, parce qu'Apple le vend depuis peu de temps avec plus de mémoire : il est désormais livré avec... 512Ko ! Évidemment, quand il s'agit de trouver et d'acheter les puces Ram supplémentaires pour monter le GS à 1 280Ko, ce qui est le minimum pour le GS aujourd'hui, ce n'est pas la joie, vu le prix de ces puces... Dans A+ de Juillet, Gary Little propose à Apple USA une mesure simple pour augmenter la vente de GS : cesser de vendre des //e ! Un //e neuf revient en fait PLUS CHER qu'un GS neuf, mais il s'en vend toujours, entre autres parce qu'il est vendu dans une version transformable en GS moyennant... plus cher que le prix d'un GS !

Ce que Gary Little ne sait pas, c'est qu'il existe un pays dans le monde où les GS sont vendus avec 1 280Ko, où on ne vend plus de //e neufs : c'est la France. Franchement, regardez un peu les annonces de la presse dévote qui vend du compatible IBM oriental : on vous vend maintenant des machines apparemment pas chères avec... 0Ko sur la carte mère, aucune mémoire ! Demandez donc le prix pour avoir la machine avec 640Ko (la moitié de la mémoire du GS). Au prix des 41256

aujourd'hui, 1 280Ko sur la carte mère au même tarif qu'avant, c'est un cadeau ROYAL. Alors que les américains n'ont droit qu'au système et au Finder, en France on nous donne GS Write et GS Paint avec la machine. On nous a même donné AppleWorks, que nombre d'Américains continuent à acheter fort cher. J'ai pu lire et entendre nombre de critiques adressées à Apple France par des possesseurs de GS, et j'ai moi-même écrit dans Pom's quelques gentillesses à propos de certains concessionnaires. N'empêche que nous sommes encore les mieux lotis. Ce qui prouve peut-être que les Français ont bien raison d'être rouspéteurs...

Virus : enfin

Alors que toutes les autres machines avaient droit à leur virus, l'Apple // jusqu'ici restait outrageusement sain. Le premier virus pour ProDOS 8, appelé CyberAids, vient enfin d'être réalisé outre-Atlantique (voir Open Apple, qui donne aussi le remède). Ce virus s'incorpore aux fichiers SYS et finit par détruire le catalogue principal des disques ProDOS. Son auteur lui a même fait une suite, appelée Festering Hate.

Même aux USA, ce virus est très peu diffusé : prévenez cependant Pom's si jamais il atteignait nos rivages. Mais si vous avez pris les précautions d'usage (voir l'article de Pierre Demblon dans Pom's 35 et le mien dans Pom's 37 page 55), vous ne risquez pas grand chose. D'autant que la communauté des utilisateurs Apple // étant bien autre chose que ce qu'Hervé Thiriez appelle dans Pom's 37 «la troupe des consommateurs des standards industriels» (suivez mon regard...), il y a plus d'exterminateurs de virus que de disquettes infectées.

Boomerang

Conséquence : l'auteur du virus a été repéré par Apple, qui a mis ses avocats sur le coup. Avoir le service juridique d'Apple sur le dos, c'est déjà dur pour une 'Major Company' du logiciel comme Microsoft (Pom's 36 page 58), alors je plains le petit malin qui va

passer soit en juridiction civile, soit en juridiction criminelle auprès du FBI, selon ce qu'aura décidé Apple. Il est vrai que mettre des virus sur des serveurs précisément au moment où Apple lance son serveur Apple Link Individuel (excellent serveur), ce n'est pas si malin que ça...

Les gentils pirates français qui ont diffusé cet été un virus GS 'doux' (ses dégâts maximaux sont l'effacement des blocs standard 00 et 01 des disques 3'5 formatés en ProDOS, et ses auteurs ont très vite diffusé un antidote), alimentant ainsi sur quelques serveurs le délire antivirus dont j'ai parlé dans Pom's 36 et 37, feraient donc bien de se méfier. Puisque, forcément, comme tous les connaisseurs, ils lisent Pom's, qu'ils sachent que la législation française récente, conçue pour lutter contre le vandalisme informatique dans les 'institutions' (banques, etc.) est également très dure. De simples gamineries au départ (le virus du groupe truc contre la désinfection du groupe machin) peuvent être prises très au sérieux par les intérêts économiques **considérables** qui sont en jeu sur cette question du vandalisme informatique, et qui ne font pas le détail...

Non, je n'exagère pas : le GS n'est pas une console de jeux, mais une très bonne machine polyvalente qui sert AUSSI au travail (voir Pom's 37 page 63), il y est même meilleur que bien d'autres. Comme il y a des gens intelligents partout, il y a partout des lecteurs de Pom's qui travaillent à amener le patron à l'Apple Expo pour remplacer son informatique ringarde. Je viens par exemple d'apprendre qu'il existe un serveur, monté par un de nos lecteurs, qui tourne avec un //c, Cristel et un Minitel pour une association s'occupant de dons d'organes (59.2.45.45). Imaginez maintenant un GS qui contrôle un processus (production industrielle, monitoring dans un hôpital, que sais-je encore : on a souvent besoin d'une bonne définition en couleurs dans le contrôle de processus) ; un pépin survient, et le gardien qui joue à Pirates la nuit sur la machine du patron DIT que c'est sans doute le virus du groupe truc. Ou pire : C'EST le virus du groupe truc...

La guerre est-elle un jeu ?

Nos voisins de RFA, dont le pacifisme est bien connu, pensent que non, et l'Office pour les Publications pour la Jeunesse a carrément interdit la diffusion d'un certain nombre de jeux de MicroProse (ceux qui ne diffusent pas Wordstar, vous vous rappelez ?), comme F15 Strike Eagle et Silent Service. Microprose a réussi à faire lever l'interdiction pour Silent Service seulement.

Cette interdiction a au moins l'avantage d'inviter à une réflexion critique sur le contenu des jeux sur l'Apple //. Reconnaissons que si on compare le nombre de jeux informatiques consacrés à la guerre à ceux consacrés à une autre activité essentielle des hommes, le sexe et l'amour, la disproportion est flagrante (sauf peut-être... en France, eh eh). Mais, à mon avis, ce qui est le plus dangereux dans n'est pas tant de jouer à la guerre (il vaut mieux y jouer que la faire), mais l'idéologie manichéenne, binaire (les bons contre les méchants) qui règne sur nos écrans depuis les 'shooting aliens' des débuts. Qu'on ne se dise pas "c'est juste un jeu" : beaucoup trop d'Américains, grands et petits, y marchent des deux jambes.

C'est d'ailleurs pourquoi les fabricants de jeux américains, du moins les meilleurs, font aujourd'hui des jeux comme **Ultima** ou **Bard's Tale** qui ont un contenu moral, éducatif : il faut être honnête, serviable, etc. pour gagner des points. La vraie question, en fait, est celle de la **qualités** des jeux. "Boum-boum sur les méchants", c'est facile et ça rapporte gros. Le jeu qui reproduit fidèlement des situations historiques réelles, qui fait réfléchir, c'est plus difficile.

Mais précisément **Silent Service** et les autres jeux de **MicroProse** sont des jeux d'excellente facture. Regardez ce que Michel dit de *Pirates* dans Pom's 37. Relisez par la même occasion ce que Christian Piard y dit de *Flight Simulator* (mais est-ce un jeu ?). À Pom's, nous pensons que le meilleur moyen de faire reculer les jeux qui rendent bête, c'est de faire connaître et soutenir les jeux de qualité, tous les jeux de qualité (y compris ceux de MicroProse).

Apple // for ever, camarades...

Selon William Mensch, Gorbatchev aurait un Macintosh sur son bureau. C'est mieux que l'horoscope de Reagan. Apple, après s'être organisé pour la conquête du Pacifique, songe très sérieusement à répondre à l'invitation que je lui lançais dans Pom's 37 de s'occuper de ce marché-là. Avec les progrès de la détente, les mentalités changent.

C'est au point que le premier jeu philo-soviétique américain vient de sortir pour GS : il s'appelle **Tetris** et est édité par **Spectrum Holobyte**. Très belles images de paysages soviétiques, mais le jeu est une sorte de casse-tête d'adresse et de stratégie, ce genre de truc à la fois amusant et énervant, extrêmement simple, qui vous fait dire «bon attends, je recommence, le prochain coup j'y arrive», et cela pendant... des heures, des semaines, des MOIS entiers : HUIT MOIS exactement selon la légende, c'est ce qu'il aurait fallu aux cosmonautes soviétiques en orbite autour de la terre pour gagner à ce jeu piloté par leur ordinateur de bord...

Du côté de la CIA et des militaires US, on reste encore cependant très "boum-boum sur les méchants", et l'on oppose à Apple que le Macintosh contiendrait certaines technologies susceptibles d'être utilisées à des fins militaires (je suppose que certains ont dû mal comprendre l'icône bombe du Mac). Mais l'on n'a rien trouvé à opposer à l'Apple // : «Le 6502 n'a jamais été très qualifié pour tuer des gens», dit William Mensch. Et voilà pourquoi John Sculley, PDG d'Apple, déclare que la meilleure machine à vendre à l'Est est le GS.

Brain storming

Après Jean-Louis Gassée et bien d'autres personnages, après le rachat de Styleware par Claris, ça devient une manie chez Apple : dès que quelqu'un fait quelque chose de bien pour l'Apple //, allez hop, on l'embauche chez Apple, quitte à le faire venir de l'autre bout du monde. La création d'AppleLink Individuel a pompé nombre de 'cerveaux' de l'Apple //. Un

rédacteur en chef de Nibble est passé chez Apple. Il restait un pays dans le monde presqu'aussi bon que nous en matière d'Apple // : c'est l'Australie. Cela essentiellement grâce à un homme, Peter Sandys, Directeur du Marketing pour Macintosh à Apple Australie, qui y a vendu non seulement des Mac, mais quantité de //e et de GS. Lui aussi rejoint le service marketing d'Apple USA pour y vendre des Apple //. Selon lui, et selon bien d'autres, tout montre qu'un tournant a été pris chez Apple USA qui n'est plus une entreprise Mac exclusivement. Une machine qui représente un milliard de dollars de chiffre d'affaires annuel vaut bien qu'on s'en occupe.

Ccci dit, rassurez-vous, à Pom's, nous restons solidement accrochés là où nous sommes, nous ne nous laisserons pas acheter. Ou alors il faudrait un pont d'or. Ou au moins 30% de plus...

Toujours plus

WDM

William Mensch (voir Pom's 37 page 54) parle décidément beaucoup en ce moment. Ce qu'il dit est intéressant, mais quand quelqu'un parle tant, c'est... qu'il a quelque chose à vendre. Que vend-il donc ?

* D'abord, bien sûr, des microprocesseurs 6502. Si vous pensiez, comme moi, qu'Apple Computer était le principal utilisateur des 6502, vous avez perdu 15 points de vie. C'est l'Atari 2600, machine de jeux, qui est l'utilisateur principal de 6502 dans le monde de l'informatique individuelle. Et dans le monde de l'informatique industrielle ? Eh bien, Sony, par exemple, utilise entre 5 et 10 millions de 6502 par an dans ses différents produits. Le 6502 a d'abord été conçu pour les appareils de contrôle électronique, et si Wozniak en a mis un dans l'Apple //, c'est pour cette raison fondamentale que c'était ce microprocesseur-là qu'il pouvait trouver pour pas trop cher chez le marchand de composants du coin. Selon Mensch, c'est le 65816 qui a été le premier microprocesseur conçu par le Western Design Center en fonction des ordinateurs individuels.

* Ensuite, des 65816 plus rapides pour le GS, Mensch ne se contente pas d'en fournir à Applied Engineering pour son accélérateur GS (voir plus bas), il est en cheville avec deux firmes, Zip et DTP, pour fabriquer des accélérateurs avec cache sur le principe de Zip Chip pour //e.

* Enfin, et surtout, Mensch veut nous vendre des... ordinateurs. Cela va commencer 'en douceur', par une carte auxiliaire de développement munie de quatre 65832 tournant en parallèle, avec un slot relié directement à chacun. Le "Mensch Family Computer" se présentera sous la forme d'un boîtier à mettre au-dessus du GS, mais aussi éventuellement d'un Atari, d'un Macintosh ou d'un IBM PC. Puis, un jour, on pourra lui connecter directement un clavier, et un autre jour... on le déconnectera du GS.

À voir et essayer. Mais si Mensch pense qu'un micro-ordinateur, c'est juste un boîtier autour du microprocesseur, je crois qu'il partage le vertige des '80486 à 40 Mhz' qui fait tant de déçus côté IBM, et sous-estime le travail d'Apple. Même Apple d'ailleurs s'est rendu compte que parler uniquement de vitesses d'horloge ou de capacité d'adressage, ce n'est pas cela qui fait vendre des Mac II. Montrer les possibilités de la machine, si. Car l'important, c'est l'ordinateur, c'est-à-dire tout le reste, le microprocesseur se trouve chez le marchand de composants du coin. L'important, c'est aussi la qualité du logiciel. Un ordinateur ringard super accéléré (j'en connais...) faisant tourner à la vitesse turbo des logiciels poussifs, c'est un peu comme un bègue qu'on ferait parler plus vite... Or déjà le GS paraît quasi-improgrammable certains jours : alors programmer ne serait-ce qu'un système d'exploitation pour quatre microprocesseurs 65832 tournant en parallèle, il y faudra de l'aspirine...

APPLE II GS, APPLE II E, APPLE II C

RÉVEILLEZ VOUS !

IIGS: Extension mémoire GS RAM , GS RAM PLUS avec patch "Super AppleWorks" et mémoire cache. Alimentation permanente d'1ou 2 cartes: RAMKEEPER. Enfin le finder disponible en quelques secondes !

IIGS, IIE: Mémoire partitionnable sous différents systèmes d'exploitation avec RAMFACTOR. (Patch "Super AppleWorks" sur Apple II E). Alimentation permanente avec le module RAMCHARGER.
Disque Winchester 3.5" externe, BJX 20 MO compact et rapide, partitionnable sous différents systèmes. Livré avec sélecteur de programmes et utilitaires en nombre.

IIE: TRANS WARP: Accélérateur à 3.6 Mgz de la mémoire principale, auxiliaire et des accès Rom. Vitesse réglable.
Extension mémoire avec 80 Colonnes: RAMWORKS III.

IIC: Extension mémoire Z-RAM ULTRA 1, 2 ou 3, de 256K à 1 MO avec ou sans horloge et/ou CP/AM. (Z-80 B 4Mgz). Compatible IIC 128K, 125K Rom 3.5, 384K. (Patch "Super AppleWorks").

APPLE II GS COMPATIBLE MS-DOS

PC-TRANSPORTER: Vous l'attendiez... Il fait merveille ! Découvrez vite le monde PC par la lucarne de votre Apple II GS en gardant vos habitudes de travail et vos périphériques.

BREJOUX.AE
29 rue Montribloud 69009 LYON
Tél: 78.36.52.69

APPLE EXPO
Nous y serons !

APPLE II FOR EVER

Toujours plus vite !

La guerre des accélérateurs est commencée. Commençons par celui qui n'a pas d'ennemi pour le moment : la carte **TransWarp GS** d'**Applied Engineering** est annoncée comme disponible au prix de 300 dollars dans InCider de Septembre, mais sans doute pour la fin de l'année en fait. Cette carte est munie d'un 65816 à 6 Mhz au lieu des 2,8 du GS. Bien entendu, elle ne peut pas accélérer les accès disques ou l'impression, qui sont toujours ralentis à la vitesse //e , mais elle accélère de deux fois les opérations

Lexique

Cache : voir Pom's 35 page 30.

CDA et NDA : accessoires pour le GS du type de l'Alarme (CDA) et du Converter (NDA) publiés dans Pom's 35.

Compilateur: programme qui se charge de traduire en bloc un programme écrit dans un 'langage' informatique pour le faire exécuter ensuite, en bloc, par la machine.

DMA (Direct Memory Access) : accès direct à la mémoire de l'ordinateur, sans passer par le microprocesseur. Le DMA est incompatible avec le cache-mémoire, donc avec les accélérateurs genre Zip Chip, mais très peu de cartes (quelques anciennes Z80 par exemple) le pratiquent sur Apple II.

Driver : programme pour piloter les périphériques et les interfaces d'un ordinateur (par exemple l'imprimante et la carte d'interface de l'imprimante).

Firmware : programme qui, au lieu d'être implanté en mémoire vive, est installé en mémoire morte dans les Roms d'une machine ou d'une carte. C'est pratique, mais quand il y a un bug dans le programme, il faut reprogrammer une Rom et changer la puce concernée.

Interpréteur : programme qui se charge de traduire et faire exécuter par l'ordinateur en direct, une instruction après l'autre, un programme écrit dans un 'langage' informatique.

MIDI : standard de communication entre un ordinateur et les instruments de musique électronique type synthétiseur.

MS/DOS : système d'exploitation repris par Microsoft sur l'ancien CP/M pour les anciens ordinateurs personnels IBM.

NDA : voir CDA.

PAO : Publication Assistée par Ordinateur (Anglais: Desktop Publishing).

Presse dévote : presse informatique personnelle dite 'générale', mais qui est en fait à la dévotion du 'standard industriel' IBM.

Relationnelle : une base de données relationnelle permet d'avoir plusieurs fichiers ouverts en même temps, avec des relations définies entre ces fichiers, si bien que la modification d'une fiche dans un fichier entraîne automatiquement la mise à jour d'autres fiches dans d'autres fichiers. Par exemple, la facturation d'un produit à un client entraîne automatiquement la mise à jour du stock et du compte du client.

RunTime : morceau de code, toujours le même, ajouté par un compilateur à un programme compilé pour l'interfacer à la machine pendant l'exécution de ce programme. Si vous détruisez le fichier BIN "Compiler" dans le Beagle Compiler, vous avez le RunTime du Beagle Compiler.

SANE (Standard Apple Numeric Environnement) : standard pour les calculs à haute précision, fourni par Apple sous formes de routines toutes faites sur ses différentes machines et dans ses différents systèmes d'exploitation.

SCSI : standard de communication rapide entre un ordinateur et ses périphériques permettant de chaîner ces périphériques et de les rendre compatibles avec différents ordinateurs.

Shareware : mode de diffusion du logiciel à encourager. Voir Pom's 37 page 52.

Standard ProDOS : logiciel sous ProDOS 8 ou ProDOS 16, non protégé, acceptant d'être placé dans n'importe quel sous-catalogue (dossier), configurable, acceptant les Classik Desk Accessories du GS et renvoyant au Quit du ProDOS en sortie.

mémoires, entre autres celles de SlowDraw II du GS. La carte est prévue pour une vitesse de 10 Mhz : dès que le Western Design Center pourra fournir à Applied Engineering des 65816 à cette vitesse (eh, Mr William Mensch, vous parlez beaucoup, mais ça ne suffit pas), il suffira de changer le microprocesseur et un quartz sur la carte. Voir Bréjoux pour disponibilité et prix français.

Bien entendu, vous hésitez : on sait que **Zip** prépare son accélérateur GS, que **MDIdeas** va bientôt sortir le sien, qu'un jour le GS sera accéléré par Apple lui-même. Oui, eh bien même pour quelques mois, moi, ça me tente. Parce que le reste, pour le moment, ce sont des projets. Et je connais des gens qui ont vu tourner à Boston GS Works à 6 Mhz...

Sur //e et //c, en revanche, c'est la guerre ouverte : pour affronter **Zip Chip**, de **Zip Technology**, **Applied Engineering** baisse le prix du **TransWarp //e**. **Zip Chip** est enfin réellement vendu, et on trouve des tests partout. Juste quelques points à savoir sur Zip Chip : d'abord ce n'est rien d'autre que l'ancienne carte accélérateur **Speedemon** de **MCT** qui a été intégrée en une seule puce. Ensuite, ça marche, c'est configurable, tout est aussi OK qu'avec TransWarp. Les tests montrent qu'en gros, TransWarp et Zip Chip, le résultat en vitesse est le même, même si le principe (cache ou DMA) est différent : tous deux se valent, et dépassent nettement le GS actuel. Enfin, Zip Chip marche mieux avec les cartes mémoire 'adressables' pour le slot auxiliaire du //e, qu'elle accélère jusqu'à deux Mhz, qu'avec les cartes 'tous slots' au standard Apple. Enfin, une différence essentielle : la Zip Chip n'occupe pas de slot, et tourne sur //c. Mais sur une Transwarp, on peut mettre un 65802 (voir Pom's 34 page 64).

Outre la rivalité Transwarp/Zip Chip, un nouveau front s'est ouvert qui met aux prises la Zip Chip et la **Rocket Chip**, de **Bit and pieces Technology**. Sauf qu'elle tourne à 5 Mhz au lieu de 4 pour la Zip Chip, et qu'elle peut aussi accélérer la ROM des cartes d'interface dans les slots (intéressant, ça, pour certaines cartes),

cette carte ressemble fort à l'autre. Si fort que Zip Technology a intenté un procès contre Bits and Pieces. Voir Call Apple de Juillet/Août pour toutes précisions.

Les prix actuellement annoncés par les constructeurs sont de 189 dollars pour la Rocket Chip, de 129 dollars pour la Zip Chip, de 169 dollars pour la Transwarp //e chez Preferred Computing.

//c instantané

Une super carte d'extension mémoire pour //c a été montrée par **Checkmate Technology** à la Fête de Boston : non seulement c'est une carte d'extension mémoire 1 Méga, mais elle intègre une carte horloge, une copie d'écran pour tous ceux qui rêvent de l'avoir dans la Rom du //c, mais ce n'est pas l'essentiel. L'essentiel, c'est le nouveau concept de cette carte : quand vous éteignez le //c, tout l'état de la mémoire est conservé par une batterie tel que vous l'avez laissé. Rien à sauvegarder sur disque ni à recharger quand on rallume : en rallumant, on se retrouve exactement là où on en était. C'est plus qu'une extension du concept de la Speedisk, ça. Cela fait de l'ordinateur sa propre mémoire de masse. Je parie que cette formule, surtout sur les portables, aura de l'avenir.

Cage à souris

Non seulement cette satanée souris a toujours tendance à gambader hors de mon bureau, mais, les rongeurs étant le plus souvent des animaux nocturnes, elle a toujours tendance à fuir la lumière et à s'enfouir le plus profondément possible sous les papiers, documents, piles de disquettes, verres de whisky, pourtant si impeccablement rangés sur nos bureaux à Pom's.

Il y a une autre solution que de la mettre sur le dos pour lui caresser le ventre comme je le proposais dans Pom's 37 : c'est de la mettre en cage. Le **MouseTrap** de **Gered Corporation** est un truc très pratique à poser sur son bureau : dessous, une cage ou une niche pour la souris, avec un beau tapis. Dessus, un plateau tournant pour les disquettes et les verres, et un présentoir pour les papiers et le dernier Pom's.

À propos de souris, un ami nous a donné un truc pour avoir un tapis à souris pas cher. Maintenant, quand le Rédacteur en Chef arrive au bureau, plus besoin de replier en vitesse le tapis du poker : c'est celui de la souris, voyons, tu vois bien que c'est du feutre. Merci, Roger.

GS/Mac : rumeurs

Ce n'est pas moi qui le dis, c'est InCider d'Août : de nouvelles Roms (oui, encore) devraient être diffusées par Apple pour le GS en Septembre. L'avenir du GS n'inclurait pas seulement un 65832, mais une chip vidéo (VGC) capable d'afficher en 640 x 400 pixels, en quatre couleurs. C'est supérieur à la résolution du Mac Plus, et le GS a déjà le même clavier, la même souris, les mêmes lecteurs de disques, les mêmes interfaces et les mêmes câbles de liaison que le Mac Plus. Il n'a pas le même microprocesseur que le Mac, mais il a des slots... Je ne vous fais pas de dessin, mais tout le monde sait que la politique d'Apple, c'est l'intégration maximum de ses deux gammes. Ce qu'a aussi le Mac Plus, c'est un port SCSI intégré, alors qu'il faut une carte au GS actuel. Mais précisément le suivant aura un port SCSI intégré...

Ce n'est d'ailleurs pas parce qu'il va bientôt être possible de mettre un Mac dans le GS que le contraire n'aura pas lieu : dans A+, Gary Little prévoit que sous deux ans Apple mettra un Apple // dans le Mac, du moins dans un Mac avec slots...

Pour les sceptiques : essayez donc de démonter votre GS, ce que j'ai déjà fait pour diverses raisons. Le constat est celui-ci : tout est entièrement modulaire. Une carte-mère se change en trois minutes, sans rien dessouder ni dévisser. À votre avis, combien coûterait à Apple la fabrication d'une carte-mère de Mac Plus, l'ordinateur ultra-amorti, avec 0Ko de Ram, mais avec un slot d'extension mémoire type GS et une sortie vidéo ? Cette carte-mère marcherait TELLE QUELLE, directement, avec les interfaces, câbles, lecteurs, boîtier, etc. du GS. Qui a un GS a dès aujourd'hui 75% d'un Mac Plus : la réciproque n'est vraie qu'à partir du SE.

Sur la carte-mère du GS, trois puces, et trois seulement, sont dans des supports, prêtes pour un changement, les autres étant soudées : le microprocesseur, la Rom, la puce VGC pour l'affichage. Ça montre ce que c'est que de savoir construire un ordinateur... C'est un fait qu'Apple a pratiquement toujours eu une politique de mise à niveau à prix très intéressant quand il introduisait un nouveau modèle de machine. Bon, vous faites comme vous voulez, moi, j'ai fait mes provisions.

Encre fraîche, S.V.P.

Comme vous le savez sans doute, il existe de petits appareils qui permettent de réencrer les rubans d'imprimante au lieu d'avoir à racheter à chaque fois non seulement l'encre, mais aussi un ruban neuf et une nouvelle cassette. En fait, on peut certainement se dispenser de ces appareils et monter un petit bricolage soi-même qui évitera de tourner manuellement la molette de la cassette. Bref, ce qui est indispensable, c'est L'ENCRE (il s'agit d'une encre spéciale, non grasse je crois).

Ce genre de 'tuyau' peut faire économiser beaucoup à beaucoup d'entre nous. Or je suis CERTAIN qu'il y a des lecteurs de Pom's qui connaissent les spécifications de l'encre, et aussi des adresses pour la trouver. Un petit mot à Pom's, et le tour est joué. Merci d'avance. Lecteurs, conservez vos vieilles cassettes d'imprimante...

Mégabit

La crise des Ram, et particulièrement des 41256, c'est du sérieux (voir Pom's 37). C'est au point qu'un Méga de puces de 1 Mégabit (8 puces pour un Mégaoctet) à 100 nanosecondes coûtait 320 dollars, le même Méga coûtant 428 dollars en puces de 256Ko (32 puces pour un Méga), dans Infoworld du 22 Août. (Je prends des 100 nanosecondes, parce qu'il faut prévoir des microprocesseurs rapides, et que la différence de prix est négligeable). Autant dire que les cartes d'extension mémoire capables d'accepter les puces de 1 Méga DÈS LE PREMIER MÉGA, et pas sur des cartes additionnelles qui ne marchent qu'après avoir rempli la première carte en 41256, deviennent intéressantes aujourd'hui : d'autant qu'avec le même nombre de supports on peut mettre 4 Mégas au lieu d'un sur la même carte.

Voici la liste et les prix (relevés dans les revues US d'Août et Septembre) des cartes d'extension que je connais qui acceptent les puces d'un Mégabit x 1 dès le premier Méga.

— Cartes pour GS seulement :

* **MultiRam GS** de **Checkmate Technology** : 127 dollars avec 0Ko installé ;
* **GS Ram Plus** d'Applied Engineering : 579 dollars avec 1 Méga installé chez **Preferred Computing** ;
* **GS Juice Plus** d'**Applied Ingenuity** : 75 dollars avec 0Ko installé, 400 dollars avec 1 Méga ;
* **Plus Ram GS-8** de **Cirtech** : 500 dollars avec 1 Méga installé.

— Carte pour slot 1-7 au standard Apple "slinky" :

* **PlusRam-16** de **Cirtech** : 500 dollars avec 1 Méga installé. C'est la seule à ma connaissance qui accepte les un Mégabit x 1 dès le premier Méga.

Le mange-disque

Lucky Luc

Activision, qui passe de plus en plus au logiciel "pro" (Paintworks Gold), a décidé de changer son nom en celui de Mediagenic, Activision restant le nom de la série des jeux médiagéniques (il paraît que ce nom plaît aux Américains : moi, ça me fait plutôt penser à des rouleaux de papier). Du coup, notre ami Luc Barthelet, qui avait opéré en douceur le glissement transatlantique VersionSoft/Activision, quitte maintenant Activision pour travailler chez Electronic Arts.

Vous voyez ce que ça veut dire ? Non ? Eh bien, comme vous le savez, Luc Barthelet, c'est un des auteurs de GS Paint (VersionSoft/Activision), bon logiciel qui a été vite dépassé par l'un des deux meilleurs logiciels graphiques pour GS, Deluxe Paint //. Je dis l'un des deux meilleurs, parce que Luc a répliqué chez Activision avec un logiciel de même niveau que Deluxe Paint, même supérieur à mon avis, PaintWorks Gold (voir Pom's 36 et 37). Pour faire du graphisme sur GS aujourd'hui, il nous faut les deux, pour avoir l'ensemble des possibilités graphiques du GS. Mais passer de l'un à l'autre en permanence est fastidieux. Ce qu'il nous faudrait, c'est un logiciel unique qui réunisse les capacités des deux, et concilie les aptitudes de Paintworks Gold avec la programmation moins gourmande en mémoire de Deluxe Paint. Si je vous rappelle que Deluxe Paint // est un produit Electronic Arts, je suppose qu'il n'y a pas besoin de vous faire un dessin...

AppleWorks v2.1

Claris USA vient de réaliser et diffuse la version US 2.1 d'Appleworks : c'est une mise à jour (update), pas une mise à niveau (upgrade). Elle contient essentiellement des réparations de bugs (bug du Control-à dans la définition des codes d'imprimantes, bug du lancement des NDA sur GS, bug de la sauvegarde le - disque - est - plein - voulez - vous - détruire - ce - fichier, bug des numéros de page après 255, bug du buffer d'imprimante sur GS, bug des plus de 100 fichiers dans un sous-catalogue), et quelques améliorations (64 caractères acceptés dans un chemin d'accès ProDOS, recalcul intelligent dans le tableur, bureau pouvant atteindre 8 Mégas sur GS). Cette version est GRATUITE pour les possesseurs d'Appleworks 2.0.

Quand on sait que Claris tient dans la revue A+ une rubrique régulière de réponses aux utilisateurs d'AppleWorks, dans laquelle elle n'hésite pas à écrire «n'utilisez pas cette option de mon logiciel, elle est bugée», qui peut encore dire que Claris néglige AppleWorks ? Quand on sait qu'AppleWorks vient seulement de perdre la première place dans le Hit-Parade des ventes de logiciels intégrés pour ordinateurs individuels (toutes marques confondues), c'est-à-dire qu'il ne détient plus "que" 25% des parts de ce marché contre 30% à Works de Microsoft pour Mac, on se dit que dès que GS Works sera chez les revendeurs, on sait qui sera le premier.

Attention, avec Appleworks 2.1, vous devez avoir les versions suivantes des programmes d'accompagnement :

- **Beagle Bros** : version 2.0 de TimeOut lui-même, et versions 2.0 ou supérieures de toutes les applications TimeOut sauf SuperFonts, QuickSpell, Thesaurus, PowerPack, DeskTools // et TimeOut Com (nouveauté de Beagle, à propos).
- **Applied Engineering** : version 3.0 ou supérieure de l'Appleworks 2 Expander.
- **Checkmate Technology** : version 5.2 ou supérieure du logiciel pour Multiram GS.
- **Jem Software** : versions 2.0 de Patchmania, Late Nite Patches, Pathfinder.
- **PinPoint** : tous les produits PinPoint qui marchent avec la version 2.0 d'Appleworks marchent tels quels avec la version 2.1.

Divers

* **Broderbund**, après avoir publié deux disquettes de "clip-art" pour Print Shop GS (Sampler Edition et Party Edition), prépare pour 89 un **Print Shop Companion** pour GS. On parle aussi d'un Print Shop 128Ko ProDOS.
* **Springboard** a réalisé une version 1.1 de son **Publisher** (logiciel de PAO sur Apple // 128Ko, voir Pom's 36), disponible en 3,5' et 5,25'. La première amélioration était une évidence : la mise au standard ProDOS, l'abandon de la protection (pourquoi faut-il que certains éditeurs ratent tant

de ventes avant de prendre conscience que la protection est une mauvaise stratégie commerciale ?). La seconde est la transformation en option du reformatage automatique du texte qui faisait redessiner tout l'écran DHGR à chaque modification du texte. Il paraît même qu'avec un Méga de disque Ram et une Zip Chip, il serait PRESQUE utilisable. Après tout, il n'est pas plus lent que le recalcul d'un fichier Multiplan moyen sur IBM PC...

* **WordBench**, diffusé par **Addison-Wesley**, et réalisé par le **Bank Street College** (Bank Street Writer) est un logiciel qui veut servir d'outillage complet pour celui qui écrit : module de prise de notes, d'esquisses, correcteur, imprimeur, dictionnaire de synonymes, etc. À voir, mais j'ai bien l'impression qu'AppleWorks/TimeOut fait aussi bien.

* **GS Works** : nous avons pu jouer avec une version béta. Une grande surprise : pas de relais-disque, sur notre GS 1280Ko, il se charge TOUT ENTIER D'UN SEUL COUP EN MÉMOIRE ! Confirmation : Claris prévoit un prix très amical pour les possesseurs d'AppleWorks (100 dollars au lieu de 250). Il va peut-être falloir sortir nos originaux d'AppleWorks du coffre-fort.

* **Art and Film Director**, d'**Epyx**, est un programme qui combine dessin et animation. À voir. Mais Epyx s'y connaît pas mal dans l'animation de l'Apple //...

* **Pages**, le logiciel de PAO de **PinPoint Publishing**, aura en même temps une version GS, une version Mac, et une version IBM. Avec GS OS, le passage de fichiers sera sans problème.

* **Pinpoint** prépare aussi un super-tableur pour GS : **Digit**.

Sous le capot

Nos Roms

En attendant qu'Apple USA décide tout comme Apple France de cesser de vendre des //e et //c, tout ce que nous pouvons faire, c'est la liste de ce que nous souhaiterons mettre dans les Roms de ces machines. Certains commencent déjà à programmer, ils ne perdent pas leur temps.

Rappelons d'abord que Pom's lui-même a déjà publié dans son numéro 23 le source d'un désassembleur de 65C02 pour le moniteur du //e, par Yvan Koenig.

> Pour quelques lecteurs qui comprennent mal mon insistance sur les lecteurs 3'5, deux citations :
>
> 1] «À l'évidence, les jours du disque 5'25 sont comptés» (A+ de Septembre) ;
>
> 2] «Le disque 5'25 est en train de subir le sort de la cartouche 8 pistes pour magnétophones» (Jon Simonsen, Beagle Bros).
>
> No comment.

Pour le //e, on souhaite surtout, comme Gérard Mourier, un dump d'écran sur imprimante : cela doit être assez simple, dans la mesure où il suffirait d'interfacer avec une carte d'interface graphique douée de ces possibilités. Pour le //c, on envisage aussi le dump d'écran, et la modification du Smartport pour qu'il reconnaisse les lecteurs Apple 3.5.

Pouce, on imprime

Tel est manifestement le rêve de nombre de nos lecteurs : avoir un ⌘-H permanent en Rom. On trouve beaucoup de bruit fait en ce moment autour de deux cartes pour GS qui permettent cela : la **Fingerprint GS** de **Thirdware** et la **Prograppler** d'**Orange Micro**. Alors, distinguons bien : la Fingerprint n'est pas une carte d'interface, c'est une carte d'interruption qui utilise les interfaces existantes (slot intégré ou carte) du GS pour imprimer. La Prograppler au contraire est une carte d'interface parallèle avec un logiciel de "Screendump" sous forme deCDA.

Franchement, je ne saurais conseiller aucune des deux. La Fingerprint fait le travail d'un CDA, mais avec du logiciel en Rom. La Prograppler est avant tout une carte d'interface parallèle. En fait, on a simplement besoin sur GS d'un CDA type Diversi-Hack ou d'un pseudo-CDA type The Desktop Manager qui fasse pour les écrans graphiques ce qu'ils font pour le texte. Pour le moment, j'utilise TDM pour sauver l'écran graphique dans le disque Ram, et j'imprime ensuite. Le GS n'a pas besoin de cartes d'interruptions, puisqu'il intègre cette possibilité sous forme de CDA.

Sur //e en revanche, une carte d'interruption est une idée intéressante. Et Fingerprint existe aussi sur //e. Mais dans ce cas, on ne voit pas pourquoi les interruptions devraient se limiter à la recopie d'écran. Il y a bien d'autres choses à faire quand on interrompt. Ce type de carte d'interruption multifonctions, avec accessoires, a existé, et existe peut-être encore sur //e. Si des lecteurs préfèrent investir dans ce type de carte que dans une unité centrale de GS, qu'ils me le fassent savoir.

La bosse des maths

Vous avez remarqué ce paradoxe : la tâche pour laquelle votre ordinateur est le moins doué, c'est... le calcul ! Le 6502 de l'Apple //, ainsi que le 65816 du GS, ne savent même pas multiplier ni diviser. Quand vous voulez une précision importante, et plusieurs chiffres après la virgule, c'est possible, mais c'est leeeeent... J'ai cru un jour que mon GS était mort : non, il calculait avec ses outils (SANE). Le 68000 du Mac fait un peu mieux seulement. La solution a donc été trouvée par Apple sur le Mac // : un coprocesseur arithmétique (68881) qui relaie le processeur principal dès qu'il s'agit de calculs.

Pour ceux qui trouvent le Mac // un peu cher, une autre solution a été trouvée pour 200 dollars : une carte coprocesseur avec ce même 68881 qui se met dans un slot de tout Apple // qui a des slots. **The Floating Point Engine** a été montré par **Innovative Systems** à la fête Apple // de Boston et il est en vente dès maintenant. L'auteur est en train de réaliser une interface pour le tableur d'AppleWorks.

Mais cette carte sera tout particulièrement utile sur GS : Innovative Systems est en train de réaliser un outil pour la Boîte à Outils du GS qui remplacera purement et simplement l'outil SANE du GS. C'est la beauté de la programmation par outils : sans rien changer aux applications qui utilisent cet outil, la vitesse de leurs calculs sera multipliée par 30...

Joystix

«C'est la faute à ce satané joystick...» : ce prétexte dérisoire ne pourra plus excuser vos scores lamentables depuis qu'**Epyx** diffuse l'outil pour gagner dans ses jeux, le **joystick 500XJ**. Pour droitiers seulement, c'est le moyen de mettre enfin votre nom en haut des "Hall of Fame".

SOS Apple //

Mr Loiseleux, de SOS Computer, le sorcier de la mécanique Apple // qui croule sous vos appels depuis qu'il s'est fait connaître dans Pom's 37, nous communique le truc suivant. Si la pile de votre GS se vide (elle est conçue pour durer six ans, mais certains GS mettent les bouchées doubles) :

1] Exigez de votre concessionnaire un changement de pile et rien d'autre : c'est une petite intervention de SAV officiellement répertoriée et bon marché, qui consiste à dessouder la pile et à en souder une autre. Mais on a entendu parler de tentatives de *changement complet de carte-mère* (!)...

2] Si vous êtes pressé, un peu bricoleur, et que votre GS n'est plus garanti, sachez qu'on peut toujours — provisoirement — remplacer cette pile par une pile ordinaire 4,5 V du commerce, type lampe de poche, après avoir dessoudé la pile Apple. Le GS de Mr Loiseleux a deux fils qui sortent du boîtier et une jolie pile extérieure...

Patchworks

Copy II Plus

Le "Filer" de **Copy II Plus** (Util.System) a pensé aux propriétaires de II Plus : écran 40 colonnes en majuscules etc. Voici quelques patches pour ceux qui ont une configuration un peu plus récente, et qui utilisent comme moi ce Filer indépendamment des Bitcopieurs (avec Prosel par exemple). Attention, ces patches sont à opérer sur la version 8.3 de Copy II Plus, celle que Pom's diffuse.

1] Quitte automatiquement vers le Quit du ProDOS, sans proposer d'autres options (c'est-à-dire renvoi automatique à Prosel par exemple) :

```
$ CALL -151 ↵
* BLOAD UTIL.SYSTEM, T$FF,
  A$2000 ↵
* 7163: 2C ↵
* 7169: F0 16 D0 14 ↵
* BSAVE UTIL.SYSTEM, T$FF,
  A$2000 ↵
```

2] Les deux patches suivants sont inutiles pour les propriétaires de GS (votre configuration est reconnue). Mais on peut sans GS avoir quand même une horloge et une carte 80 colonnes. Notez qu'il faut toujours répondre Oui à la proposition des catalogues en 80 colonnes, pour avoir des arbres convenables pour les sous-catalogues.

—pour que Copy II Plus accepte automatiquement la date et l'heure de ProDOS :

```
$ CALL -151 ↵
* BLOAD UTIL.SYSTEM, T$FF,
  A$2000 ↵
* 2241: 2C ↵
* 2244: EA EA EA EA ↵
* BSAVE UTIL.SYSTEM, T$FF,
  A$2000 ↵
```

— pour avoir automatiquement les catalogues en 80 colonnes :

```
$ CALL -151 ↵
* BLOAD UTIL.SYSTEM, T$FF,
  A$2000 ↵
* 23D3: 2C ↵
* 23D6: F0 1A ↵
* BSAVE UTIL.SYSTEM, T$FF,
  A$2000 ↵
```

3] L'option COPY du menu de Copy II Plus vous propose en premier le BITCOPY 5.25 et le BITCOPY 3.5. Si vous choisissez une de ces deux options, il lance soit le fichier BITCOPY.SYSTEM, soit le fichier BC3.SYSTEM. Mais je me sers moins souvent de ces deux copieurs (de toute façon parfaitement lançables directement par un sélecteur), que de Prosel par exemple. Il suffit de trouver (avec les bitcopieurs précisément) les deux noms de fichiers cités dans UTIL.SYSTEM, et de les remplacer par d'autres noms de fichier (précédés d'un octet de longueur) pour que Copy II Plus vous renvoie automatiquement au fichier que vous avez choisi.

Ultima V

Ultima V, de Broderbund, a des musiques magnifiques sur //e avec cartes Mockingboard, Phasor, Passport ou équivalent. Mais si vous avez mis ces cartes dans votre GS, elles ne sont pas reconnues. Voici un patch diffusé par les éditeurs pour y remédier :

```
10 FOR I = 768 TO 702 : READ A
   : POKE I, A : NEXT : CALL 768
20 DATA 24, 251, 194, 48, 244,
   11, 0, 162, 3, 35, 34, 0, 0,
   225, 244, 4, 0
30 DATA 244, 0, 0, 244, 0, 3, 1
   62, 3, 16, 34, 0, 0, 255, 56,
   251, 226, 48, 96
```

Lancez ce programme par RUN et lancez aussitôt Ultima V par PR£6. Vous pouvez aussi renvoyer votre disquette, l'échange est gratuit.

Print Shop Haute Densité

A+ d'Août publie un truc tout simple pour avoir une impression de bonne qualité avec Print Shop, en lui faisant passer deux fois la tête d'impression. Mettez un ruban noir sur votre imprimante, mais déclarez à Print Shop (la dernière version, couleur), dans le menu de configuration, que vous avez un ruban tricolore. Après avoir fait votre carte de vœux, mettez toutes les bordures, fontes, ainsi que les graphiques en vert ou en orange. Pour faire du vert, Print Shop imprimera une fois en bleu, une fois en jaune, c'est à dire... deux fois en noir. Ce truc doit marcher avec tous les logiciels qui impriment du graphisme en couleurs.

Sauvegarde

Comme je viens seulement d'y penser, je présume que certains n'y ont pas

pensé non plus. En lecteur averti, vous savez qu'il faut toujours faire des doubles. Quand vous travaillez avec AppleWorks par exemple, vous sauvez deux fois vos fichiers, sur votre disquette données et sur son double. Donc, soit vous passez votre temps à retirer à remettre des disquettes, soit vous jonglez avec les menus pour changer de lecteur. Temps perdu. Il suffit (c'est ça le truc) de désigner à AppleWorks votre disque de données comme un Catalogue ProDOS (/POMS/NUMERO.38) et non comme slot et drive (Lecteur 2 Port 6), de mettre vos deux disquettes dans deux lecteurs différents, de laisser chaque disque dans son lecteur, et de se contenter d'éjecter ou d'ouvrir la porte de l'un des deux. ProDOS trouvera automatiquement le bon, ⌘-S marchera à tout coup, et vos disquettes ne traineront plus.

Encore une partie et j'éteins

In Door

Le jeu, ce n'est pas seulement "boum-boum sur les méchants". **Superstar Indoor Sports**, de **Mindscape**, vous permet de prendre plaisir à des simulations très bien agencées de ping-pong, de fléchettes, de bowling. On choisit son niveau, on peut affronter l'ordinateur ou un copain. Mais ce qui m'a le plus surpris, c'est d'ailleurs le meilleur jeu des quatre : un "air hockey" (hockey sur air) auquel je n'ai jamais joué "en vrai". Formidable, et ça coûte moins cher qu'un voyage en Californie pour y jouer.

Allô, Michel...

— Attends, le détecteur me signale un radar, et je dois doubler ce camion avant le haut de côte. Où est la quatrième sur ce bazar ? Ouais, ça vaut pas la Lamborghini. Bon, l'overdrive maintenant.

Comment ça, pas d'overdrive sur ce truc ? Mais c'est un char d'assaut, celle-là ! Bon, je suis marron, les flics dans le rétro.

Le jeu de ce numéro, ce sera **Test Drive**, d'Accolade : en gros, tu es le journaliste de l'Auto Journal, et tu testes les super-bolides européens dont nous rêvons tous : Ferrari Testarossa, Lotus Turbo Esprit, Porsche 911 Turbo, Lamborghini Countach, tu peux y aller. Un circuit routier bien raide, tu les pousses au bout de leurs possibilités, mais sans sortir, sans te planter (attention au rétroviseur), sans couler une bielle (attention au compte-tours). Tu as potassé le dossier technique avant, avec courbes d'accélération et tout. Formidable. Mais, soi-disant pour faire "réellement international", ils ont mis la Corvette de Chevrolet. Tu parles, UN VEAU, oui. Bon, je prends la Lotus ce coup-ci...

— Salut Michel, teste bien. Avant son démarrage, Michel a oublié de nous dire qu'il testait la version //e : formidable telle quelle, mais si bon sang ils pouvaient y mettre le SON du GS...

Pour une poignée de dollars

Dans la série «À ce prix-là c'est idiot de pirater", c'est **Accolade** cette fois-ci qui vend ses jeux *favoris de tous les temps* à 15 dollars pièce. Cela comprend des jeux comme **Hardball** et **Ace of Aces**, qui en valent la peine.

Virus égyptien

Je suppose que, comme tout le monde, vous avez un jour été atteint par un des virus les plus dangereux sur Apple //, le virus **Lode Runner**. Vous croyez peut-être qu'à force, vous êtes immunisé. Eh bien, essayez donc **La revanche du Pharaon** (Pharao's Revenge) de **Publishing International** : moi, j'y ai reperdu pas mal d'heures de sommeil. *Laisse cette disquette ici, bon sang. Non, je ne joue pas avec. J'étudie ce jeu pour les lecteurs de Pom's : je TRAVAILLE, moi !*

Apple // planant

Vexant, l'article de Christian Piard sur Flight Simulator Mac dans Pom's 37, quand on sait qu'il n'y a pas jusqu'à présent de version GS de ce jeu. Mais dans un prochain Pom's, Christian va apprendre à piloter des turbo jets avec un général de l'armée de l'air US, cassera du bois en hélicoptère, et testera pour vous le véhicule spatial de combat et d'assaut le plus avancé du 21e siècle, le C104. Et le tout sur Apple //, cette fois...

16 bits

Hyper GS Card

Si **Tutor Tech**, prédécesseur Apple // d'Hypercard pour le Macintosh (cf. Pom's 36 page 60), ne vous suffit pas parce qu'il n'exploite pas le graphisme du GS, vous devriez trouver votre bonheur avec le produit "hyper-like" que **Roger Wagner** prévoit de publier en Septembre et de présenter à la fête de l'Apple // de San Francisco (oui, la deuxième fête Apple // de l'année : côte Est, côte Ouest. Et nous, en Europe, on s'y met quand ?).

Mais Pom's vous prévient charitablement : nous n'avons pas plus l'intention de diffuser des disques domaine public de piles "hyper GS" que de commencer à copier les piles de centaines de disquettes de piles Hypercard pour le Mac qui s'entassent ailleurs. Ça s'empile, ça s'empile, ça nous horripile. Pouce, on pile !

Copie d'écran

Pierre Demblon a oublié de regarder sa disquette d'utilitaires Diversi avant de répondre au lecteur qui lui demandait un ⌘-H permanent : non seulement il n'est pas impossible de savoir où le GS range l'écran-texte en cas d'accès aux CDA, puisque la documentation de **Diversi-Hack** le dit, mais ce même Diversi-Hack vous fait une copie d'écran impeccable. En plus, il reconnaît les caractères souris et les caractères en inverse de l'écran et les imprime correctement sur Imagewriter II, ce que ne fait pas **The Desktop Manager**. Il permet aussi d'envoyer à l'imprimante toutes les commandes qu'on veut. Enfin, ce Diversi-Hack est sur la disquette Shareware "DPA.0003" diffusée par Pom's...

CDA de CDA

En plus, Pierre ne vous a pas tout dit sur **The Desktop Manager**, de **On Three**. Ce *Canada Dry du CDA*, selon

sa formule, est capable d'interrompre (et donc d'imprimer les écrans)... des CDA eux-mêmes. Bon, j'explique : du moment que l'application que vous utilisez autorise les interruptions, vous pouvez à tout moment l'interrompre pour utiliser un accessoire CDA. D'accord ? Si ce CDA lui-même autorise les interruptions, il n'y a pas de raison théorique pour qu'un deuxième CDA ne puisse pas interrompre le premier, d'accord aussi ? Simplement, ce n'est pas possible pour les CDA orthodoxes, parce que pour avoir accès au second CDA, il faut passer par le menu des CDA, ce qui n'est possible qu'après avoir quitté le premier CDA. Mais justement, TDM n'est pas un CDA orthodoxe. Le seul des CDA que j'utilise qui n'autorise pas les interruptions, c'est... le tableau de bord GS.

Bref, le GS est bien fait : il accepte des interruptions en cascade, sans planter. J'ai vu pas mal d'Apple // : c'est apparemment le seul modèle qui n'ait pas trop de problèmes avec les interruptions. Il va falloir bûcher sur ces fichiers $B6.

Yvan Koenig nous signale qu'il existe une nouvelle version de TDM qui fait disparaître les "ù", mais qui introduit un bug dans le presse-papiers. La solution est pourtant simple, si nos amis américains voulaient bien nous écouter : utiliser les caractères Mousetext comme un STANDARD, par exemple pour les encadrements en texte. Les caractères Mousetext (sauf deux) sont les mêmes sur tous les Apple // du monde.

Fatal System Error --> 0911

C'est un fait : un utilisateur de GS sur 3 dans le monde a droit de temps en temps, et de façon imprévisible, à ce message sur son écran. Avec un peu de chance, et si vous disposez d'un sort de mage de niveau 7, ça doit repartir au bout d'un petit moment.

Quantité de spéculations ont été faites sur cette erreur. Apple, jusqu'à présent, fait celui qui ne sait rien, arguant qu'il n'arrive pas à la reproduire VOLONTAIREMENT. C'est pourtant bien simple : il suffit de taper CALL —454 (sur un GS avec les Rom 01). Cela plaiderait pour un bug dans le firmware.

Le responsable, selon **Computist** d'Août, serait plutôt le contrôleur ADB (l'erreur 911 est une erreur ADB), puce n° 19 soudée sur la carte mère. Quoi qu'il en soit, hard ou soft, IL Y A UN BUG dans un GS sur trois. Conclusion : chaque fois que vous avez cette erreur, donnez un coup de fil à Apple Assistance. Et si votre GS est sous garantie, envoyez un mot à votre concessionnaire, pour être certain que la réparation sera gratuite le jour où Apple aura trouvé un remède.

Si jamais vous voyez un de vos lecteurs chaînés sur votre GS démarrer de façon intempestive, et que vous ne pouvez l'arrêter qu'avec Reset, il est probable que votre lecteur Apple 3.5 est fautif. Votre concessionnaire doit gratuitement vous remplacer sa carte interne avec une carte Révision B. Si vous avez une carte d'extension mémoire pour GS avec un code de date 8751, 8752 ou 8801, il est possible que certaines puces Ram marchent mal. Même chose : voir votre concessionnaire, et Open Apple d'Août pour toutes précisions.

Musique

* La carte stéréo/digitalisation (numérisation) des sons **Sonic Blaster** d'**Applied Engineering** est en vente chez Preferred Computing au prix de 99 dollars. Elle permet d'enregistrer, éditer, et rejouer les sons du magnétophone ou de la chaîne. Son amplificateur intégré fait qu'il n'est pas nécessaire de la brancher sur l'ampli de la chaîne : elle accepte directement les enceintes sans ampli.

* Il faudra la comparer à la carte **Digitizer Professional**, de **Mdideas**. Plus chère (190 dollars), et exigeant la carte stéréo **Supersonic** de la même maison, sur laquelle elle se branche, cette carte prétend à des caractéristiques "pro", par exemple une réduction du bruit de fond.

* La carte **Audio Animator**, toujours d'**AE**, fait la même chose, mais avec les instruments MIDI. Environ 200 dollars chez Preferred Computing. Voir Bréjoux pour la disponibilité et le prix en France de ces deux cartes.

* **Music Studio** Version 2.0, qui marche entre autres avec les cartes ci-dessus, est disponible chez **Activision/Mediagenic** (voir adresse à Music Update), pour 10 dollars plus le port si vous renvoyez en même temps la page 1 du manuel de l'ancienne version. Vingt instruments nouveaux, 15 caractéristiques nouvelles, dont la possibilité d'entrée en temps réel par un interface MIDI.

Trou de mémoire

Sean Nolan, dans Call Apple de Juillet, révèle un emplacement extrêmement précieux dans le GS, un endroit libre où vous pouvez mettre ce que vous voulez : un trou dans la mémoire. Sean fait partie de ces programmeurs ancienne mode qui aiment bien savoir où est le programme qu'ils ont fait. La méthode idéologiquement correcte, recommandée par le Bureau Politique, pour gérer la mémoire du GS, c'est de passer par le Memory Manager. Mais entre les pointeurs de pointeurs et les déplacements perpétuels, plus moyen de savoir où on est.

Il y a bien certains emplacements que le Memory Manager ne gère pas, ceux réservés à l'affichage texte et graphique dans les bancs E0 et E1. C'est le registre de "shadowing" (ombre portée, explication dans un prochain Pom's) en $C035, qui détermine l'usage de ces zones (en gros : sous ProDOS 8, le GS copie ce qui se passe dans les bancs mémoire 00 et 01, dans les bancs E0 et E1). Ce registre a huit bits qui sont autant de commutateurs pour le shadowing. Mais on en a oublié un : celui de la page texte 2. Alors, Apple a fait par soft ce qu'il n'a pas fait par hard, pour permettre aux rarissimes programmes qui, dans les musées, utilisent encore cette page texte 2, de tourner : c'est l'Alternate Display Mode du Tableau de bord. Celui-ci, par interruptions, copie tous les 60èmes de seconde, la zone $800 à $BFF du banc 00 dans le banc E0. Bon. Mais ils en ont encore oublié un : la page texte 2 de la mémoire auxiliaire. Je ne connais pas de programme 8 bits assez tordu pour faire de l'affichage en utilisant la page texte 2 en 80 colonnes, mais s'il y en a un, alors il est incompatible avec le GS.

Si vous n'avez rien compris aux explications super-denses qui précèdent, vous n'êtes pas le seul. Mais la conclusion est là : tant que de nouvelles Roms n'auront pas récupéré cet espace, nous pouvons mettre ce que nous voulons en $E1/800 - SE1/BFF dans le GS, le Memory Manager n'y touchera pas, cet espace est LIBRE. Enfin, il l'était avant ce numéro de Call Apple, car il va sans doute faire comme la page $300 - $3CF dans l'Apple // : TOUT LE MONDE va y mettre des routines fixes 'indispensables', puisque c'est libre.

Système 3.2 : officiel

La version 3.2 du système que j'évoquais dans Pom's 37 est sortie de la clandestinité en Juillet aux USA. Versions : ProDOS 16 1.6, ProDOS 8 1.6, Basic System 1.2, Finder 1.1, System Utilities 3.1. Essentiellement des bugs réparés, dans les outils également. Dans le sous-catalogue /SYSTEM/SYSTEM.SETUP, gardez TOOL.SETUP et TS2 si vous avez les Roms 01, TS1 si vous avez les Roms 00; tout le reste peut être jeté si vous n'êtes pas branché Appletalk. Même chose pour le driver IMAGEWRITER LQ dans /SYSTEM/DRIVERS qui ne permet que l'émulation d'ImageWriter par l'intermédiaire d'Appletalk. Le driver d'ImageWriter est, lui, tout nouveau, et en nets progrès.

Gros efforts sur le son : drivers d'interface MIDI, et de nouveaux outils. A.C.E (Audio Compression and Expansion), pour comprimer et décompresser les fichiers de sons numérisés. Outil MIDI. Note Sequencer (séquenceur de notes) pour enregistrer et réécouter les sons MIDI. Note Synthetizer (cette fois, je ne traduis pas) qui fabrique des sons et des notes de musique.

Comme d'habitude, ce nouveau système devrait être disponible "dans quelques temps", gratuitement, chez votre concessionnaire. Comme d'habitude, vous avez plus de chances de l'avoir, avec sa vraie documentation, à l'APDA ou à DDA, pour un prix modique. Si vous n'avez pas tout compris dans les deux paragraphes qui précèdent, lisez l'article de Pierrot Demblon dans Pom's 36.

En vrac

* Certains programmes de la disquette DPA 002 comme P8CDA connaissent tant de succès en Shareware qu'on en envisage, paraît-il, de nouvelles versions à diffusion commerciale classique. Si, après essai des versions Shareware, certains de ces programmes vous conviennent, écrivez à l'auteur pour savoir s'il en existe des versions commercialisées.

* J'ai trouvé, lors de tentatives (guère fructueuses) pour ranger un peu les piles de disquettes à Pom's, un accessoire (NDA) bien utile : sauvegarde sur disque d'écrans SHGR, copie sur imprimante de ces écrans, redéfinition du clavier pour les caractères nationaux spéciaux. Copyright Apple : merci à nos amis d'Apple... Allemagne. Ce "Tastaturdienst", ou "KeyUtil", devrait pouvoir être traduit en Français, nicht wahr ?

* Pour dessiner sur GS (et aussi sur Mac), un truc génial (cher, mais génial) : la tablette graphique ADB (s'intercale entre le clavier et la souris) **Kurta is/one** de **Kurta Corporation**. Revue complète dans A+ d'Août.

À lire

La loi de la vie

Disparition de deux revues sur Apple // ces derniers temps : **Apple User**, revue d'Outre-Manche, consacrait peut-être un peu trop de place à des vieilleries (CP/M etc.). Elle passe ses lecteurs à **Apple 2000**, revue genre Pom's (bonne revue, donc) du groupe d'utilisateurs le plus important d'Europe.

Assembly Lines, la petite revue ronéotypée que Bob Sander-Cederlof consacrait à son assembleur S-C Assembler, disparaît : plus personne ne se sert de cet assembleur, on ne connaît plus que Merlin et Orca. Bob, excellent programmeur de firmware, excellent désassembleur d'AppleWorks (voir Pom's 37 page 64), s'est fait embaucher par... Applied Engineering bien sûr.

Si certains meurent, d'autres naissent ou se développent : des revues comme **Open Apple**, **A Plus**, **Pom's**... gagnent des lecteurs. Une revue "underground" se crée pour les fanatiques de l'Apple //, **The Road Apple**. Non que les possesseurs d'Apple // soient plus nombreux qu'avant, mais ils lisent de plus en plus, ils sont fidèles à leur machine et aux revues... sur l'Apple : quand Call Apple a voulu ouvrir une rubrique MS/Dos, elle a cessé tout de suite à cause des résiliations d'abonnement (dont le mien). Et quand Computist a voulu ouvrir elle aussi une rubrique MS/Dos, elle ne trouve rien du tout à mettre dedans, sauf des lettres de protestation de lecteurs. Apple // For Ever...

En plus, il semble que la tendance des lecteurs, très sensible à Pom's, à demander aux revues Apple // indépendantes un SERVICE croissant (Hot Line, vente de logiciels et de matériels...), s'affirme aussi aux États-Unis : après la vente de livres, Open Apple se met maintenant à vendre du hard, sous la forme des cartes d'extension mémoire de Cirtech, que Tom Weishaar appelle l'Applied Engineering européen, ce qui est une juste comparaison.

Woz

Interview de **Steve Wozniak** dans **InCider** d'Août. À lire de près. Et ne demandez plus d'où viennent mes opinions sur l'avenir de l'Apple //.

Tricher n'est pas jouer

Si, si, moi je pense que c'est de la triche. Même s'il est diffusé par un éditeur de jeux ; même s'il est rédigé par une bande de fanatiques hagards qui ont l'aventure dans le sang et qui, eux, sont allés jusqu'au bout à la force du poignet ; même si leurs indices sont codés pour que tu ne puisses pas lire la solution tout de suite; même si c'est un florilège de leur excellente revue ; même si tu ne le regardes qu'au bout du désespoir, avant ton dernier souffle ; même s'il ne contient que des indices, et pas la solution toute faite. Même, c'est de la triche.

Dis donc, je suis complètement perdu dans **Breakers**. Y-aurait pas le plan de Borg, dans **Quest for the clues** ? Comment ça, de la triche ? Mais si tu

n'avais pas causé, j'aurais pu le finir moi-même, ce plan ! Non mais, moi, moi qui lis régulièrement **Quest-Busters**, le journal des aventuriers, moi, UN TRICHEUR ???

Tiens, regarde plutôt à la fin, il y a la liste des meilleurs jeux pour les aventuriers, et celle des machines sur lesquelles on peut jouer à ces jeux. Ils y sont tous, l'IBM, le ST, le Mac, le C128 et le reste. Et quelle est la machine qui les a TOUS, sauf un ? OK, tu viens de gagner 20 points d'expérience.

Adresses

A Plus
P.O. Box 40158, Philadelphia, PA 19106-9931, USA.

Accolade
20813 Stevens Creek Boulevard, Cupertino, CA 95014, USA.

Activision
(attention, nouvelle adresse) :
3885 Bohannon Drive, Menlo Park, CA 94025, USA.

Activision Music Update
P.O. Box T, Gilroy, CA 95021-2249, USA.

Addison-Wesley Publishing Company
6 Jacob Way, Reading, MA 01867, USA.

APDA
290 S.W. 43rd Street, Renton, WA 98055, USA.

Applied Engineering
P.O. Box 5100, Carrollton, TX 75011, USA.

Applied Ingenuity
14992 Ramona Boulevard, Unit M, Baldwin Park, CA 91706, USA.

Assembly Lines
S-C SOFTWARE, P.O. Box 280300, Dallas, TX 75228, USA.

Beagle Bros
6215 Ferris Square, Suite 100 San Diego, CA 92121, USA.

Bits and Pieces Technology
Gantry Entreprises, P.O. Box 2622, San Rafael, CA 94912, USA.

Bréjoux
29, rue Montribloud 69009 Lyon — ☎ 78 36 52 69

Broderbund Software, Inc.
17 Paul Drive, San Rafael, CA 94903, USA.

Call Apple
290 S.W. 43rd Street, Renton WA 98055, USA.

Check Technology
509 South Rockford Drive, Tempe, AZ 85281, USA.

Computist
P.O. Box 11086-T, Tacoma, USA.

Cirtech
Currie Road Industrial Estate, Galashiels, Selkirkshire, Scotland, TD1 2BP, Royaume-Uni. ☎ 0896 57790.

DDA/Prim'Vert
36, rue des États Généraux 78000 Versailles.

Electronic Arts
Electronic Arts Direct Sales, P.O. Box 7530, San Mateo, CA 94403, USA.

Epyx
600 Galveston Drive, Redwood City, CA 94063, USA.

Gered Corporation
2401 Northeast Cornell Road, Suite 125, Hillsboro, OR 97124, USA.

Imagol
72, bd Raspail, 75006 Paris. ☎ (1) 42 22 05 55.

Incider
80 Elm Street, Peterborough, NH 03458, USA.

Innovative Systems
P.O. Box 444, Severn, MD 21144-0444, USA.

Jem Software
P.O. Box 20920, El Cajon, CA 92021, USA.

Kurta Corporation
4610 South 35th Street, Phoenix, AZ 85040, USA.

Mdideas
1163 Triton Drive, Foster City, CA 94404, USA.

Mediagenic
voir Activision.

Memoland
2, boulevard Bellevue 05000 Gap — ☎ 92 51 03 30.

Mindscape
3444 Dundee Road, Northbrook, IL 60062, USA.

On Three
4478 Market Street, Suites 701-702, Ventura, CA 93003, USA.

Open Apple
P.O. Box 11250, Overland Park, Kansas 66207, USA.

Orange Micro
1400 N Lakeview Avenue, Anaheim, CA 92807, USA.

Pinpoint Publishing
5901 Christie Avenue, Emeryville, CA 94608, USA.

Preferred Computing
P.O. Box 815828, Dallas, TX 75381, USA.

Publishing International
333 West El Camino Real, Suite 222, Sunnyvale, CA 94087, USA.

QuestBusters
P.O. Box 525, Southeastern, PA 19939-9968, USA.

Roger Wagner Publishing
1050 Pioneer Way, Suite P, El Cajon, CA 92020, USA.

Scarlett
Big Red Computer Club, 423 Norfolk Avenue, Norfolk, NE 68701, USA.

SOS Computer
50, rue Rochechouart 75009 Paris — ☎ (1) 42 81 03 73

Springboard Software
7808 Crekkridge Circle, Minneapolis, MN 55435, USA.

Spectrum Holobyte
Sphere, Inc, 2061 Challenger Drive, Alameda, CA 94501, USA.

The Road Apple
1121 N.E. 177th Street, Portland, OR 97230, USA.

Thirdware Computer Products
4747 NW 72nd Avenue, Miami, FL 33166, USA.

Western Design Center
2166 East Brown Road, MESA, AZ 85203, USA.

Zip Technology
11926 Santa Monica Boulevard, Los Angeles, CA 90025, USA.

Remords

Le MEM-monde

∞ Remords, c'est le mot, à propos du MEM-monde. J'appelle ainsi ce monde de la programmation ensoleillée — niçoise — qui a toujours eu — à tort — sur Apple// un caractère un peu marginalisé, un peu à part, comme les adeptes du CP/M. À Pom's, nous n'y sommes pas très compétents. C'est pourquoi jusqu'ici Pom's n'a pas suffisamment dit tout le bien qu'il en pensait. Nous faisons donc un appel au peuple : lecteurs utilisateurs de MemDOS, MemBasic, MémoBase et les autres, auteurs de logiciels du MEM-monde, écrivez-nous pour nous dire ce que vous en savez et ce que vous en pensez. En attendant, voici deux ou trois choses que je sais du MEM-monde.

D'abord, quel est le premier interpréteur (pas compilateur) Basic 16 bits qui a été disponible pour le GS ? Réponse : MemBasic. Excellent produit, qui pourrait (à tort) un peu rebuter au premier abord les fanas de l'Apple, à cause de son 'look IBM' : il y a une raison bien précise, et fort louable, à ce look. C'est que le même produit existe sur d'autres machines, avec échange possible des programmes et des données. MemDos sait par ailleurs ce qu'ignore GEOS : comment protéger ses produits sans gêner l'utilisateur. MemBasic est protégé par une mini-carte électronique 'en béton' (incassable), qui se met dans le slot 4 du GS, celui dans lequel on ne peut rien mettre, carte qui n'a pas à être déclarée dans le tableau de bord. Les disquettes, elles, sont au standard ProDOS. Sécurité parfaite pour les données (car **seules nos données** sont importantes). C'est 'pro', ça.

Ensuite, quelle est la première (et pour le moment la seule) base de données relationnelle pour GS en 16 bits ? MémoBase, bien sûr, qui, elle aussi, existe sur d'autres machines. Un RunTime existe pour que vous puissiez diffuser vos bases en MémoBase. Interface avec MemBasic. Un souci évident de l'essentiel, la sécurité de vos données (sauvegarde automatique et autres raffinements). Ça aussi, c'est 'pro'.

Ce que je vous dis là vient très largement d'un développeur en MemDos, le Docteur Walusinski, qui m'a envoyé ses produits en MemDOS sur disque 5"25. Attention, ces produits n'ont PAS besoin de la carte MemDos pour tourner sur //e, //c et IIGS. Les disquettes sont non protégées et sous 'Turbo-MemDOS' (et pourquoi le MemDOS de ma carte n'est-il pas turbo, lui ?). Je suis un peu dubitatif sur "MemCalories" : très bonne base de données diététique pour médecins, mais aussi pour particuliers qui veulent garder la ligne, très accessible et rapide. Le 'hic', ce sont les chiffres exorbitants que me renvoie cette base de données : ce n'est pas possible qu'il y ait tout ça dans l'inoffensive bouteille de Chivas de Pom's 34...

J'apprécie donc mieux "Biblio", outil de recherches bibliographiques et documentaires très commode et bien fait. Tout fonctionne par menus déroulants, mais en écran texte 80 colonnes, la formule que je préfère. Marche sur //e, //c et IIGS, y reconnaît les disques 3"5 et les disques durs.

Mais bien entendu, celle que je préfère, c'est celle qui ne considère pas le Bordeaux 47 comme une somme de calories, mais pour ce qu'il est : un avant-goût du paradis. Il s'agit de "MémoCave", logiciel de gestion de votre cave à vins. On sent que ce programme a été fait par un connaisseur. Et s'il est largement au niveau des logiciels américains du point de vue de la programmation, il les ridiculise carrément sur l'essentiel : les catégories de vin de MémoCave n'ont rien à voir avec le classement des piquettes 'compatibles' californiennes. MemDOS et MémoCave : vive la France ! Californiens, continuez à faire des ordinateurs...

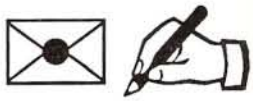
Deux adresses pour le MEM-monde : MémoLand et Imagol.

∞ *Sottisier* : vu dans Science et Vie Micro (SVM) de Septembre : «Le GS, un ordinateur du passé». Lu le même jour sur le serveur SVM en 3615, rubrique "Choisir un Micro" : «Apple IIGS, **256Ko**, à partir de **17 430 F** neuf». On se surpasse, dans la presse dévote...

• Tilt, revue de jeux sur ordinateurs, vient, lui, de découvrir l'existence du GS. Il était temps...

• Tetris : 4655, dernier score de Michel, sur son ordinateur du passé 256Ko à 17 430F.

Courrier des Lecteurs



Kruptos

Philippe Picherit

À chaque essai de la commande externe de cryptage Kruptos (Pom's 29), j'obtiens un Syntax Error sur mon Apple //e 65C02. Que faire ?

Le rédacteur devait être un peu crypté et vous en êtes victime ! En fait, la commande de cryptage est : CRYPTE chemin d'accès, clef [,Sslot] [,Ddrive] au lieu de KRUPTOS chemin d'accès...

Daisy Wheel Printer

Michel Imbert

Existe-t-il sur Macintosh un Traitement de Textes, un Tableur et un Gestionnaire de Base de Données fonctionnant avec l'imprimante à marguerite Apple ?

Word, traitement de textes peut utiliser votre imprimante. Excel, parmi les tableurs s'en servira également. Nous ne doutons pas que les gestionnaires de BdD Microsoft s'accomodent de vos marguerites mais il vous faudra poser la question aux revendeurs.

Caractères MouseText

Michel Anceaux

Comment obtenir les caractères MouseText à l'écran d'un Apple //e ?

Il vous faut mettre votre Apple //e à niveau (65C02 et nouvelle Roms). Consultez votre revendeur.

ImageWriter II ordinaire ?

Daniel Fieux

Quel programme permettrait une utilisation optimale de mon ImageWriter II par mon Apple //c ? Existe-t-il des logiciels de fontes pour améliorer le résultat ordinaire ?

Dans ce numéro, Pom's vous propose LA solution : AppleWorks + TimeOut SuperFonts qui est livré avec plus de 40 polices et qui donne un résultat identique à celui du Mac. Si vous ajoutez nos nouvelles disquettes Polices 01, 02 et 03, vous voici avec près de 150 polices.

Affitext

Fabienne Dunaigre

Pourquoi le message No Buffer Available avec Affitext du Pom's 36 ?

Question encore très fréquente : réponse en encadré dans ce même numéro 36, page 17.

Copy II Plus

Dominique Entremont

Je possède Copy II Plus américain. Faites-vous les mises à jour, à quelles conditions ?

Non. Central Point Software pratique directement les mises à jour de ses logiciels.

Détecteur d'appels

Marc Lerman

Votre revue est super, mais... êtes-vous certains que le plan du détecteur d'appels du n° 32 soit correct ?

Oui, il va vous falloir contrôler à nouveau votre montage.

ProDOS

Freddy Cozette

Comment stocker des informations dans la carte langage sous ProDOS ?

Vos informations OU ProDOS, il faut choisir : ProDOS occupe en effet la carte langage. Dans le célèbre et excellent Beneath Apple ProDOS, vous trouverez une solution pour forcer le chargement de ProDOS en mémoire principale comme DOS 3.3. Cette solution présente toutefois de menus inconvénients comme l'incompatibilité avec Basic.System...

Renumber sous ProDOS

François Vignes

Existe-t-il un Renumber sous ProDOS ?

C'est peut-être là la question la plus souvent posée. Il figure sur la disquette livrée avec le livre La Programmation en Basic avec ProDOS édité par Inter-Éditions, disponible chez les revendeurs Apple.

CalvaCom

Didier Bonnet

Quel est le moyen de s'abonner à CalvaCom, le numéro publié dans Pom's ne semble pas être le bon ?

Le numéro de Calva a changé : Télétel 1 par le 3613 puis 1751111. Au code d'abonné, répondez : 'nouveau'.

Adresses

Didier Skutnik

Sauriez-vous me donner des adresses pour AppleWorks GS et PaintWorks Gold ?

Activision 3885 Bohannon Drive, Menlo Park, CA 94025 USA
Claris 10, rue Mercœur 75011 Paris
✆ (1) 43 70 71 72

Graphisme sur Minitel

Claude Metzger

Pour sauvegarder le source DENSITE.S, quelle est la syntaxe exacte : BSAVE DENSITE.S, A$???,L$???

Le source, comme tous les sources est à saisir à l'aide de l'éditeur d'un assembleur ou avec un traitement de texte avant assemblage. Ici, le mieux est d'utiliser Merlin-Pro ; l'option de sauvegarde est au menu.
Pour utiliser ce programme de graphisme sur Minitel, vous n'avez pas besoin d'utiliser le source : le programme Basic et la récapitulation DENSITE sont les seuls indispensables.

Stockage des chaînes

Robert Sevrin

J'ai tenté, avec Block.Warden de Prosel, de franciser les noms des mois dans le fichier ULTRA.SYSTEM de TimeOut UltraMacros, version américaine.

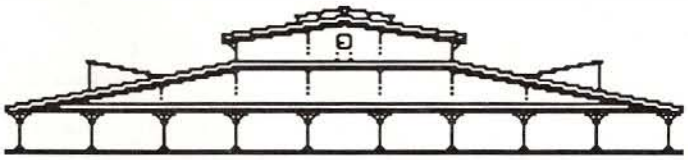
La liste des noms se présente comme suit, sans espaces entre les différents mois :
JanuaryFebruaryMarchAprilMay...

Changer April en Avril n'offre pas de difficultés et ⌘' donne bien Avril 8,88.

Mais quand il s'agit de changer September en Septembre, pourtant du même nombre de lettres, la commande ⌘' donne SeptembreOctober 8, 88.

Il en est de même à chaque fois que je modifie le dernier caractère d'un mois. Où est l'erreur ?

Sept bit sur huit suffisent pour coder tous les caractères de l'Apple //. Le bit de poids fort (MSB) n'est pas signifiant pour ce codage : l'espace peut être codé indifféremment $20 (00100000 en binaire) ou $A0 (10100000). Le 'A' est codé $41 (01000001) ou $C1 (11000001).
Bien des programmes utilisent ce 8ème bit comme délimiteur de chaîne : tous les caractères sont codés avec le MSB à 0 sauf le dernier à 1. Ou bien l'inverse.
Regardez en hexa votre chaîne de caractères : tous les caractères ont un code inférieur à $80 sauf le dernier de chaque mois. Ou bien l'inverse.
La solution consiste donc à respecter le bit de poids fort lors de vos modifications. En effet la routine d'affichage du logiciel se sert de ce MSB pour détermier la fin des mois...



Apple Expo 1988

Grand rendez-vous de la communauté Apple, Apple Expo ouvrira ses portes du mercredi 28 septembre à 10 h au samedi 1er octobre à 19 h à la Grande Halle de la Villette, Porte de Pantin à Paris. C'est pour vous l'occasion d'un tour d'horizon de l'actualité Apple. C'est aussi l'occasion de rencontrer Pom's :

Éric Weyland vous accueillera durant les quatre jours.

Jean-Yves Bourdin, incollable sur le GS et l'actualité Apple, répondra à vos questions le samedi de 15h à 17h.

Sylvie Gallet (rappelez-vous le compacteur de programmes Basic, le formateur de programmes, les fractales en Pascal, Xcat l'indispensable commande externe en Orca/M, Alarme l'accessoire GS...) sera également présente le samedi de 15h à 17h.

Dimitri Geystor ? Avec un peu de chance, vous le croiserez dans la journée de samedi. On ne présente plus ce spécialiste d'AppleWorks et de SuperMacroWorks qui participe aujourd'hui aux TimeOut sur AppleWorks 1.4.

Venez nombreux (gardez Pom's à la main, vous passerez les caisses économiquement et pensez à votre carte Joker).

Sommaires :

Recueil 1
(Pom's 1, 2, 3 & 4)
Overlay dynamique
Visicalc et Applesoft
Un programme aide-mémoire
De l'ITT 2020 à l'Apple
Les adresses du graphique
Présentation graphique
Graphiques haute-résolution
Création de tables de formes
Des instructions en une lettre
À s'arracher les cheveux
PLE : Program Line Editor
CRAE : Applesoft Editor
Les mystères du DOS
Inverseur DOS 3.2/DOS 3.3
Les drames de l'Append
Réparez votre Append
Réparez votre Renumber
Analyse du contenu des slots
Le nettoyage-mémoire
Faites le ménage en mémoire
Utilitaires de documentation
Limites des éditeurs de textes
Copie d'écran texte
Catalogue général en Pascal
Formattez vos programmes
La leçon de calcul
Trois secondes pour trier
Chargez vite les fichiers bin.
Sprechen Sie DOS ?
Changez votre poignée de jeu
SHLAM : routine bien pratique
Apprendre l'assembleur
Déplacement des programmes
Notions de base : les fichiers
Print Using d'intérêt général
Conversion Pascal/Basic
Communication grâce à l'Apple
Le format DIF
Les fichiers EXEC
Un exemple de Hello
Personnalisez vos disquettes
TRACE sélective
Un programme devinette
Les mémoire de masse
La carte M/DOS à l'essai
Les codes ASCII épluchés
Survol de l'Apple ///
RobotWar

Recueil 2
(Pom's 5, 6, 7 & 8)
HAIFA : amper interpréteur
Le clavier magique
Création graphique en Pascal
Logiciel graphique en Pascal
Graphique, tu nous tiens...
Tortue &
Graphiques et logique
Les Quatre Ponts
Hard Copy Seikosha
Création de polices
Francisez le DOS
La programmation facilitée
Un analyseur de syntaxe
Programme de test universel
Programme de Hello complet
Accélérez vos programmes
Des programmes relogeables
Recherche de codes binaires
Ergonomie des programmes
Applesoft vers EXEC
Création de fichiers EXEC
Tableaux en Pascal
Dump Pascal
Un générateur
Routine d'input généralisé
Gestion de fichiers
Gestion de masques en Basic
Boot PLE+CRAE
Programme de fondu enchaîné
Le Hbasic : un basic pascalien
Calculs en format gestion
Calculs à 12 chiffres en Pascal
Le loto, c'est facile
Cryptographie à clef publique
Visicalc et textes
Arcanes du moniteur ///
Moniteur étendu
L'Apple //e à l'essai
Le cours de Basic Applesoft
Base de données sur Apple
Mini-base de données
Pilot et Super-Pilot à l'essai
Multiplan à l'essai
Banc d'essai du BASIS 108
Allo, Questel ?
The Last One à l'essai
C.O.R.P. à l'essai
CX Multigestion à l'essai
Test de la carte Legend 128
La souris de Lisa
FID, MUFFIN et DEMUFFIN

Recueil 3
(Pom's 9, 10, 11 & 12)
Copie GR d'écran HGR
Un éditeur graphique HGR
Fusion de tables de shapes
Fondu enchaîné graphique
Reconstituez le Puzzle
La Magie de Magicalc
Éditeur-Composeur de texte
Caractère à votre imprimante
Super-impression de chaînes
Mise en forme de listings
Lecture de fichiers TEXT
Saisie Multipage en Pascal
Fichiers avec RWTS
Opcodes d'assembleurs
La Prom P5A désassemblée
Jonglez avec votre catalogue
MatGraph, routine graphique
Compression d'images HGR
Dessins avec planche à clous
Aide au graphique HGR
Caractères géants
Décisionnel graphique à l'essai
Commandes automatiques
Gestion de comptes bancaires
Accès direct aux disquettes
Édition des fichiers Basic
Coupure de fin de programme
Input généralisé de tableaux
Chargement de l'Integer
De Big Mac vers Lisa 2.5
La communication avec l'Apple
CX Système à l'essai
Calendrier en Visicalc
Music
Initiation à l'assembleur (1)
Le Bavard
Effets stroboscopiques
Pour supprimer les REMs
Des POKEs à gogo
Un programme de menu
Un menu à la carte
Comparaison de programmes
"Type-ahead" en Applesoft
Courbes en conversationnel
Extremum absolu
Tri rapide en assembleur
MacArticle
Gutenberg à l'essai
Traitement de textes
Magic Window à l'essai
Initiation à l'assembleur (2)
Disque virtuel 16Ko
Charts Unlimited à l'essai
Le Calculateur Entier
Exp(π¢Sqr(163)) entier ?
Patch du DOS 3.3
Tableaux de dimension variable
& ONERR GOTO
Assembleur à la fin du Basic
Nombres flottants en LM
PEEKs et POKEs en Pascal
Les codes ESC de votre PLE
Tracé rapide de cercles
Un éditeur de shapes
Trucs pour AppleWriter
AppleWriter 1.1 et l'Apple //e
Pom's a vu MEM/TERM
Analyse de MagicMailer
La famille Apple //
L'Apple //c est né

Pom's 13
Initiation à l'assembleur (3)
Home, Sweet Home
Jeu en Pascal : Ordralphabetix
ProDOS à l'essai
La compatibilité de l'Apple //c
Analyse de la VTOC
Bloc-Notes
Impressions des variables
Tables de shapes
Réduction d'images HGR
PEEKs à gogo
Le Basicium
Lecteur Micro-expansion
BugByter à l'essai
Pom's a vu LIGHT 1
ThinkTank à l'essai
Les nouvelles du Macintosh

Pom's 14
Initiation à l'assembleur (4)
Procédure &
Disque virtuel 64Ko
Lecture/écriture de tableaux
Input généralisé miniature
Amper-interpréteur ICARE
Les essais du Macintosh
De MacWrite à AppleWriter
Les tokens du Basic Microsoft
Gestion de masques améliorée
Lève-toi et brille !
217 fichiers par disquette
Conversions minus/MAJUS
Un catalogue général
Koala Pad à l'essai
Appleworks à l'essai
Visicalc Advanced Version
Jane, Apple// sauce Macintosh

Pom's 15
Initiation à l'assembleur (5)
MobbyDisk
Disk Check-up
Fleuves de France
HPGRAPH
Écriture en page HGR
MousePaint
Caractères programme sur IW
Fichiers avec Rwts et Dos 3.3
Routines en ROM du Macintosh
Formes et curseurs
Omnis 2 à l'essai

Pom's 16
Initiation à l'assembleur (6)
Pot-pourri ProDOS
Gérer la date en ProDOS
Signature
Epson en mode proportionnel
Accès direct aux disquettes
L'Apple en multitâche
Personnalisez vos disques Mac
Call : un exemple d'application
MacPaint et le Basic
Appel des routines en ROM
Entrée analogique améliorée
Désassembleur en Basic/WPL
Les codes ASCII du //e
Startup, saisie de date Pascal
PLE+CRAE+APA
Conversion minus/MAJUS
CALLs à gogo

Pom's 17
Initiation à l'assembleur (7)
Transfert rapide de tableaux
Les pointeurs en Pascal UCSD
Animations graphiques
Distribution normale
Copie d'écran sur EPSON
Catalogue sur imprimante
BSAVE et BLOAD en Basic
Switch vidéo pour carte 80 col
Mini-éditeur Basic
Transformez votre //e en][+
Conversion chiffres-lettres

Pom's 18
Disquette DOS/ProDOS
Saisie de variables par écran
Le porte parole
Gestion de fenêtres
Graphiques aléatoires
Récupérez les icônes système
Informations pour le Finder
Les curseurs du système
Micro-journal
Les pirates ont la parole...
Hard-Copy HGR étendue
Les pointeurs, suite...
Rendez le DOS transparent
Recherche dans un tableau

Pom's 19
Un analyseur de syntaxe

du recueil 1 au Pom's 37

Un catalogue multi-sed
Une mémoire tampon d'écran
Disquettes DOS/Pascal
Mousecat
Lucy in the Sky with Diamonds
Étrange accessoire
Fermez les fenêtres
Où est la souris
Développement 68000
La méthode PERT
Hard-copy IIGR IW
L'intelligence artificielle
Vos programmes perdus
Création de caractères

Pom's 20

DOS ou ProDOS à la carte
Manipulation de catalogue
Copie d'écran sur silentype
Quatre fonctions pour le Basic
Des messages en boîte
Gestion de comptes bancaires
Décalages...
Utilisation de la carte langage

Pom's 21

Conversion Pascal/DOS
Édigraph
Courrier automatisé en WPL
RWTS désassemblée
Décrypteur
Library ; motif de satisfaction
Une corbeille dans la pomme
Fenêtres de saisie
La 36ème piste
Un compatible incompatible ?
Défilement graphique
Calcul de π

Pom's 22

Composition de n°de tél.
Adaptation à ProDOS
Compacteur de Basic
Copie d'écran GR
De la formule à la courbe
'Startup' Basic sous ProDOS
Des titre en Basic
Jeu de mots
Transfert // —> Mac
Hyperparallélépipèdes
MacAstuces
Mini-éditeur Basic
Six Lettres
Des bugs sous ProDOS
Print Using

Pom's 23

Le Turbo Pascal
Un composeur sur le //c
GENUTILE
Formateur de listings Basic
Navette
Un "Décruncher"
Un Désassembleur 65C02
Jeu de la Vie
Routine de saisie en Basic
MacAstuces
Impôts sur Multiplan
Impôts sur Multiplan
SoftCopie
Retour dans le Basic
Halte aux Scrolls
Patches au DOS 3.3

Pom's 24

Un disque virtuel sous CP/M
THGR sur DMP et IW
EXECSAVE
Printer
Datheur
TOKENisation des chaînes
PostScript
Composeur téléphonique Mac
Macintosh et Presse-Papiers
Le Bundle et le Finder
Analyse multicritère
Hard-Copy sur GP500A
Le kit 65C02
Snake

Pom's 25

Temps anglais
Éditeur de blocs ProDOS
Miroirs Janus
Reset & ProDOS
Compdisk
Formateur ProDOS
Ad litteram
Ascenceurs en Basic
MacAstuces
µFinder
Tri de chaînes
Disk][et //c
Courbes fractales
Hyper-espace

Pom's 26

Des fenêtres en HGR
Disquettes sans DOS
Renommez les cdes ProDOS
Startup Pascal complet
Carte Super Série et CP/M
Tri de chaînes
Accessoire : 68000's
Courbes
Loupe HGR
CALLRWTS
Compter.animaux
Lissajous

Pom's 27

TéléAlarme Télématique
Un carnet d'adresse
Éjectez les disquettes 3,5"
L'apprenti-FORTHier
Carte SSC et CP/M (2)
Mise en page
MacAstuces
EssaisMac
Minitel et Macintosh
Le Minitel
Minitel et Apple //
L'Apple IIgs
Un crayon optique

Pom's 28

SHOW, I, C, O
InterPom's Apple //
CopyBasFiles
Messages et Basic.System
Comment utiliser Pom's
Essais Mac
InterPom's Mac
JustText à l'essai
Scrollings en HGR
Extasie à l'essai
Pico-Expert

Pom's 29

Recherche d'octets
Caractères Vidéotex
Conversions HGR/DHGR
Kyram : de l'octet au Nibble
ProDOS et fichiers détruits
Cryptage : Kruptos
Accessoire : Kruptos
PageMaker : essai 'routier'
Macintosh SE et II
Simplexe
AppleWriter et /RAM
Tabulations automatiques WPL
Ligne tél. artificielle

Pom's 30

GSPaint et Basic
La souris ? facile !
Humeur...
MultiScribe 2.0
DhgrTool : outil graphique
Jeu : souris-Laser
'Patch' de BugByter
Freeware : un billard à l'essai
Disquettes Pom's Mac B à I
Essai Mac : More 2
Mac et le II : T_Pom's
Apple // et le II : T_Pom's

Pom's 31

Peeks et Pokes longs sur GS
Le Basic et l'heure
Graphes et fonctions
GOGO
Essai : carte vocale Valentine
Écriture en DHGR
Éditeur Vidéotex Pom's
Essai Mac : Writer +
Décompacteur Basic
Accessoire : MinAcc
Kino
Nouveaux InterPom's
Le voyageur
Graphisme en Turbo-Pascal
LeMouse : souris et menus

Pom's 32

HGR -> SHGR
Courbes fractales
Écran virtuel
Un détecteur de sonnerie
Essai : Unimate
Démarrer sur Unidisk 800Ko
Commande XCAT
Des filets en WPL
Essai Mac : Microsoft Works
Programme 'Mots-croisés'
Fuseaux horaires
Essai Mac : Orthogiciel 2
Copy : une commande externe
COPIE.TF
Pascal : fichiers séquentiels

Pom's 33

La date avec AppleWriter
Graphisme sur Minitel
Raccourcir WPL
GS et QuickDraw II
Heure et AppleWriter
Commande externe : ONLINE
Impression avec AppleWriter
Cats et Mouse
Mac : en vitrine
Graphisme et Minitel
Essai Mac : SideKick 2
Recherche d'applications
NEWBOOT
Essai : SuperMacroWorks
AppleWorks et LaserJet+
Multiscribe 2.0

Pom's 34

Copie d'écran texte 40 et 80
AppleWorks : macros
Le passage des paramètres
À l'essai : les TimeOut
72 points avec MacWrite
Essai Mac : Overvue 2.1
Pom_Link 2.1
Répondeur télématique
Du côté des LaserWriter

Pom's 35

Multi-tâches : Share
CDA : Alarme
Courrier automatique
Petit guide ProDOS
NDA : Converter
Compilateur Beagle
Big U de Beagle
Accessoire : Corner light
Mac : en vitrine
Accessoire : Mini-montre
Double-Graphics

Pom's 36

Vivre sans ProSel
HGR : Affitext
ProDOS : comment faire ?
Le CD-Rom Apple
Formes en DHGR : Sprite 1.0
Essai : Xpress et graphisme
SuperMasterMind
SMW : Almanach et Mailing
BRAM

Pom's 37

Travaux pratiques Ensoniq
Une image, un source : Cut
APW : macros commandes
Programmer en AppleWorks
Copie de lignes avec SMW
ProDOS : des indispensables
AppleWorks : un nouveau 'look'
Les nouveaux écrans ProSel
(Mac)Plus deflèches
Accessoire : Quid?
Essai : Flight Simulator
Les Copy II
Sharewar e: disquettes DPA

En *italiques* : articles Mac
= article d'intérêt général

Bon de commande

Gagnez du temps ! Avec votre Minitel et de votre carte de crédit, appelez notre serveur gratuit 24 heures sur 24 au (1) 39 53 04 40.
Pour les envois par avion, ajoutez 15 F par numéro et/ou disquette, câble.

➡ Revues Pom's

n° 8 35,00 F ☐	n° 10 40,00 F ☐	n° 11 40,00 F ☐	n° 12 40,00 F ☐	n° 13 40,00 F ☐
n° 14 40,00 F ☐	n° 15 40,00 F ☐	n° 16 40,00 F ☐	n° 17 40,00 F ☐	n° 18 40,00 F ☐
n° 19 40,00 F ☐	n° 20 40,00 F ☐	n° 21 40,00 F ☐	n° 22 40,00 F ☐	n° 23 40,00 F ☐
n° 24 40,00 F ☐	n° 25 40,00 F ☐	n° 26 40,00 F ☐	n° 27 45,00 F ☐	n° 28 45,00 F ☐
n° 29 45,00 F ☐	n° 30 45,00 F ☐	n° 31 45,00 F ☐	n° 32 45,00 F ☐	n° 33 45,00 F ☐
n° 34 45,00 F ☐	n° 35 45,00 F ☐	n° 36 45,00 F ☐	n° 37 45,00 F ☐	n° 38 45,00 F ☐

➡ Disquettes Pom's Apple][, 140Ko – 5,25 pouces

Ces disquettes regroupent l'ensemble des programmes pour Apple // d'une revue.

n° 1+2 60,00 F ☐	n° 3 60,00 F ☐	n° 4 60,00 F ☐	n° 5 60,00 F ☐	n° 6 60,00 F ☐
n° 7 60,00 F ☐	n° 8 60,00 F ☐	n° 9 60,00 F ☐	n° 10 60,00 F ☐	n° 11 60,00 F ☐
n° 12 60,00 F ☐	n° 13 60,00 F ☐	n° 14 60,00 F ☐	n° 15 60,00 F ☐	n° 16 60,00 F ☐
n° 17 60,00 F ☐	n° 18 60,00 F ☐	n° 19 60,00 F ☐	n° 20 60,00 F ☐	n° 21 60,00 F ☐
n° 22 60,00 F ☐	n° 23 60,00 F ☐	n° 24 60,00 F ☐	n° 25 60,00 F ☐	n° 26 60,00 F ☐
n° 27 60,00 F ☐	n° 28 60,00 F ☐	n° 29 60,00 F ☐	n° 30 60,00 F ☐	n° 31 60,00 F ☐
n° 32 60,00 F ☐	n° 33 60,00 F ☐	n° 34 60,00 F ☐	n° 35 60,00 F ☐	n° 36 60,00 F ☐
n° 37 60,00 F ☐	n° 38 60,00 F ☐			

➡ Disquettes Pom's Apple][, 800Ko – 3,5 pouces

Ces disquettes regroupent l'ensemble des programmes pour Apple // d'une revue.
Elle sont au format ProDOS.

n° 29 80,00 F ☐	n° 30 80,00 F ☐	n° 31 80,00 F ☐	n° 32 80,00 F ☐	n° 33 80,00 F ☐
n° 34 80,00 F ☐	n° 35 80,00 F ☐	n° 36 80,00 F ☐	n° 37 80,00 F ☐	n° 38 80,00 F ☐

➡ Disquettes Pom's pour Macintosh

Ces disquettes regroupent l'ensemble des programmes pour Macintosh d'une revue. Programmes du domaine public, polices de caractères, nouveaux systèmes etc. y sont joutés. Détails dans chaque numéro.

n° 14+15+16 150,00 F ☐	n° 17 80,00 F ☐	n° 18 80,00 F ☐	n° 19 80,00 F ☐	
n° 20 80,00 F ☐	n° 21 80,00 F ☐	n° 22 80,00 F ☐	n° 23 80,00 F ☐	n° 24 80,00 F ☐
n° 25 80,00 F ☐	n° 26 80,00 F ☐	n° 27 80,00 F ☐	n° 28 80,00 F ☐	n° 29 80,00 F ☐
n° 30 80,00 F ☐	n° 31 80,00 F ☐	n° 32 80,00 F ☐	n° 33 80,00 F ☐	n° 34 80,00 F ☐
n° 35 80,00 F ☐	n° 36 80,00 F ☐	n° 37 80,00 F ☐	n° 38 80,00 F ☐	

➡ Recueils de la revue Pom's

Ces recueils regroupent quatre numéros de Pom's.

n° 1 (revues 1 à 4) 140,00 F ☐	n° 2 (revues 5 à 8) 140,00 F ☐	n° 3 (revues 9 à 12) 140,00 F ☐
Disquettes 1 à 4 200,00 F ☐	Disquettes 5 à 8 200,00 F ☐	Disquettes 9 à 12 200,00 F ☐

➡ Câbles communication

Pour la mise en œuvre des câbles, se reporter aux numéros 27, 28, 30, 31, 33 et 34 de Pom's qui présentent les programmes de communication.

Câble Minitel/ordinateur 225,00 F ☐ Précisez le type d'ordinateur _____
Câble local ordinateur/ordinateur 225,00 F ☐ Précisez les types _____ / _____

Envoyez ce bon et votre règlement à : Éditions MEV – 12, rue d'Anjou – 78000 Versailles

Nom : _____

Adresse : _____

Règlement par : CB/Visa/Euro/MasterCard ☐ Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte ____ ____ ____ ____ date d'expiration __ __ __

Montant ____,__ F Signature :

Bon de commande

Gagnez du temps ! Avec votre Minitel et de votre carte de crédit, appelez notre serveur gratuit 24 heures sur 24 au (1) 39 53 04 40.
Pour les envois par avion, ajoutez 15 F par numéro et/ou par disquette soit, par exemple, 90 F pour un abonnement avec ou sans disquettes.

➡ Logiciels pour Apple][

Ludologic 140Ko	80,00 F ☐	Ordico 140Ko	200,00 F ☐	E.P.E 5.1 140Ko	200,00 F ☐
E.P.E5.1 800Ko	200,00 F ☐	Clv_Pom's 140Ko	200,00 F ☐	Clv_Pom's 800Ko	200,00 F ☐
InterPom's 2.0 140Ko	450,00 F ☐	InterPom's 2.0 800Ko	450,00 F ☐		
Pom_Link 3.1 140Ko	450,00 F ☐	Pom_Link 3.1 800Ko	450,00 F ☐ *(avec carte Joker : 400,00 F)*		

SuperMacroWorks	500,00 F ☐	*(Joker : 450,00 F)*	Compilateur Beagle	750,00 F ☐	*(Joker : 675,00 F)*
Big U	450,00 F ☐	*(Joker : 400,00 F)*	ProSel	500,00 F ☐	*(Joker : 450,00 F)*
Copy II Plus 8.3	500,00 F ☐	*(Joker : 450,00 F)*	PathFinder	150,00 F ☐	
TimeOut Graph	850,00 F ☐	*(Joker : 765,00 F)*	TimeOut SuperFonts	800,00 F ☐	*(Joker : 720,00 F)*
TimeOut UltraMacros	650,00 F ☐	*(Joker : 585,00 F)*	TimeOut SideSpread	550,00 F ☐	*(Joker : 495,00 F)*

➡ Disquettes 'shareware' pour Apple][

Présentation des 'DPA' dans le numéro 37 et des 'Polices' dans ce numéro.

DPA 140Ko '001' ☐ '002' ☐ '003' ☐ '004' ☐ '005' ☐ '006' ☐ '007' ☐ 80,00 F par disque te
DPA 800Ko '001' ☐ '002' ☐ '003' ☐ '004' ☐ '005' ☐ '006' ☐ '007' ☐ 80,00 F par disquette
Polices 140Ko '001' ☐ '002' ☐ '003' ☐ 80,00 F par disquette
Polices 800Ko '001' ☐ '002' ☐ '003' ☐ 80,00 F par disquette

➡ Logiciels pour Macintosh

Excel efficace 400Ko	190,00 F ☐	Excel efficace 800Ko	175,00 F ☐	MacAstuces	200,00 F ☐
Clv_Pom's	200,00 F ☐	InterPom's 2.0	450,00 F ☐	Raccourci	200,00 F ☐
Le livre du Macintosh	200,00 F ☐	Pom_Link 3.1	450,00 F ☐ *(avec carte Joker : 400,00 F)*		
Copy II 7.1	500,00 F ☐ *(avec carte Joker : 450,00 F)*				

➡ Disquettes 'domaine public' pour Macintosh

Mac 'A' ☐ 'B' ☐ 'C' ☐ 'D' ☐ 'E' ☐ 'F' ☐ 'G' ☐ 'H' ☐ 'I' ☐ 80,00 F chaque

➡ Logiciels pour IBM PC

Secrets de Multiplan 175,00 F ☐ InterPom's 2.0 450,00 F ☐

➡ Reliures

Reliures toilées pour 6 numéros de Pom's (un an) : _____ exemplaire(s) à 75,00 F, soit _____ F

➡ Abonnements (comprenant la carte Joker)

La carte Joker offre un accès privilégié à notre Hotline (assistance dans la mise en œuvre des programmes diffusés par Pom's) et donne droit à des remises sur les produits Pom's.

Abonnements pour six numéros à partir du _____, à :

la revue Pom's seule	225,00 F ☐
la revue et les disquettes Apple][140Ko	525,00 F ☐
la revue et les disquettes Apple][800Ko	625,00 F ☐
la revue et les disquettes Macintosh	625,00 F ☐
la revue Pom's, les disquettes Apple][140Ko – 5' 1/4 et les disquettes Macintosh	925,00 F ☐
la revue Pom's, les disquettes Apple][800Ko – 3' 1/2 et les disquettes Macintosh	1025,00 F ☐

Envoyez ce bon et votre règlement à : Éditions MEV – 12, rue d'Anjou – 78000 Versailles

Nom : _____

Adresse : _____

Règlement par : CB/Visa/Euro/MasterCard ☐ Chèque bancaire ☐ Chèque postal ☐ Mandat ☐

numéro de la carte ____ ____ ____ ____ date d'expiration __ __ __

Montant ____, __ F Signature :

Jamais un outil aussi *rapide* n'a été aussi *simple*

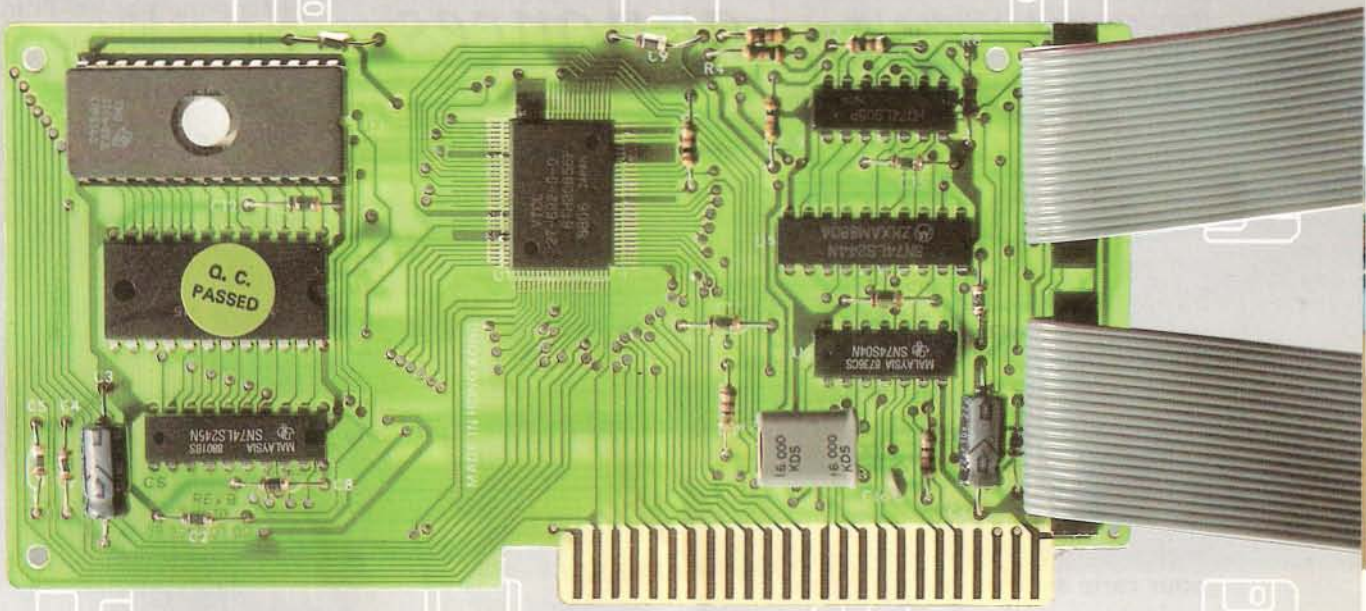
Compiler un programme Basic
avec le BON OUTIL, c'est :

- Transformer le Basic en langage machine, sous ProDOS,
- Accélérer de 2 à 20 fois l'exécution des programmes,
- Générer un code plus compact que le Basic original,
- Compiler instantanément, lors du RUN, ou sur disque,
- Utiliser les cartes d'extension-mémoire,
- Gagner en confidentialité : le code est hermétique,
- Gagner en efficacité...

Compilateur, documentations française et américaine
750,00 F TTC, (abonnés 675,00 F), frais de port 20,00 F.
Pom's – Éditions MEV – 12, rue d'Anjou – 78000 Versailles
(commande par Minitel : (1) 39 53 04 40)
Banc d'essai, Pom' 35

Compilateur Beagle™

Universal Disk Controller



...LA solution pour relier aux Apple //e et][+ :

- un "Apple 3.5 GS Platinum" 800Ko
- un lecteur Macintosh 800Ko
- un lecteur Macintosh 400Ko
- un lecteur Chinon 800Ko
- un Unidisk 5,25" 140Ko

Connectée à n'importe quel port, elle pilote deux lecteurs (même différents, pouvant totaliser 1,6 méga). Elle permet le travail bit à bit avec tous les lecteurs d'aujourd'hui (ce travail était impossible avec l'Unidisk 3.5").

Carte UDC : 1 050,00 F TTC franco
Tarif carte Pom's Joker : 945,00 F TTC

Logma S.A. – 12, rue d'Anjou – 78000 Versailles – Tél : 39 51 24 43

Petites annonces

Vends Unidisk 3.5' 800Ko avec sa carte contrôleur pour II+, //e ou IIGS : 2 500,00 F.

M. Chacon ✆ 90 74 29 64 dans la journée ou 90 04 75 59 après 19h.

Vends lecteur 3.5' 800Ko pour Apple IIGS (acheté le 16/03/88), 2000,00 F.

Thierry Biancarelli – 47, rue du Javelot – 75645 Paris Cedex 13

Vends ou **échange** contre matériels et accessoires Apple, 1 ordinateur Junior-Computer 64Ko à base de 6502, avec nombreuses améliorations matérielles et logicielles et commutations de Roms en bancs de 16Ko. Idem pour un adaptateur Tekelec Minitel vers imprimante parallèle. Idem pour carte synthèse de parole en français pour Apple II et carte digitalisation de son pour Apple II.

Gérard Mourier ✆ (1) 40 35 26 25 (soir)

Vends Apple //e 128Ko 65C02, 80 colonnes, écran couleurs, lecteur 5,25' & contrôleur, souris, joystick, livres, excellent état, 6 500,00 F. Lecteur Unidisk 3.5 avec carte & manuel, Mousedesk 3.5, 3 000,00 F.

Alain Chane To Hing – 95, route de St François – 97400 St Denis - Réunion ✆30 09 54

Recherche pour Apple //e carte interface pour disque dur Profile.

Vends serveur Minitel : Apple //e, Appletel, Profile 5 Mo, carte CP/M, imprimante, 15 000,00 F à débattre.

Patrick Bonnot – 4, Cour du Ginkgo 75012 Paris ✆ (1) 43 40 49 14 de 20h à 22h.

Vends Apple //c avec moniteur monochrome, souris, 2 joysticks, nombreux livres, lecteur externe, 5 000,00 F.

Armand Peyret-Forcade – 8, rue Bernard Palissy – 92800 Puteaux ✆(1) 47 78 75 79 après 19h.

Vends Apple IIGS couleurs 1,2Mo, un lecteur 5,25', un lecteur 3,5', 9 000,00 F.

Thomas Schmitt ✆ (1) 49 86 55 88 après 19h.

Recherche une souris avec une interface pour Apple //e, en état de bon fonctionnement.

Michel Barran ✆ 99 81 95 24

Recherche explications : Pin Point Dialer, numéroteur automatique à partir d'une base de données AppleWorks, ne fonctionne pas en France sur le réseau téléphonique. 300,00 F de prime à celui qui trouve la solution !

Michel Sallat ✆ 45 62 52 68

Recherche conseils pour faire fonctionner une carte Time II sur un Apple //e.

Philippe Dormont ✆82 25 64 95

Vends Apple //e 84, 5000,00 F.

Guy Le Peutrec ✆35 20 77 96

Solution du n° 37

A	S	S	I	S	T	A	N	T	E
R	E	T	R	O	U	V	A	I	S
T	R	E	M	B	L	E	R	A	S
D	E	P	A	R	A	■	I	R	A
E	N	P	■	E	R	O	N	E	R
V	I	E	S	■	E	R	E	■	T
I	T	■	T	A	M	I	S	E	E
V	E	N	E	R	I	E	■	R	A
R	■	A	L	I	E	N	E	R	A
E	S	T	E	S	■	T	R	E	S

Horizontalement

1 - Affolés
2 - Coéquipier
3 - Administrations
4 - Rogna - Stable
5 - Flairé - Premiers de l'édition
6 - N'en finit pas de mettre en morceaux - Vit
7 - En cours d'arrestation - Bossai
8 - Lopin - Moindre
9 - Voyelles - Organisme agricole
10 - Situé - Subsistas

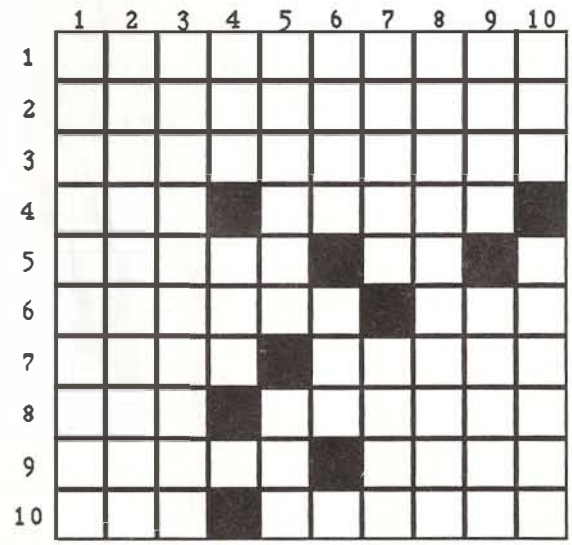
Verticalement

1 - Adopterais
2 - Recouvrirai
3 - Préparées
4 - Pour prendre l'air ! - Pot d'alchimiste
5 - Arrivait - Fossé
6 - Grains odorants - Choix
7 - Panier percé - Base d'apéritif
8 - Franchement, non !
9 - Des ans - Apprendra
10 - Personnel - Détendis

Problème 38

par

Joëlle Piard



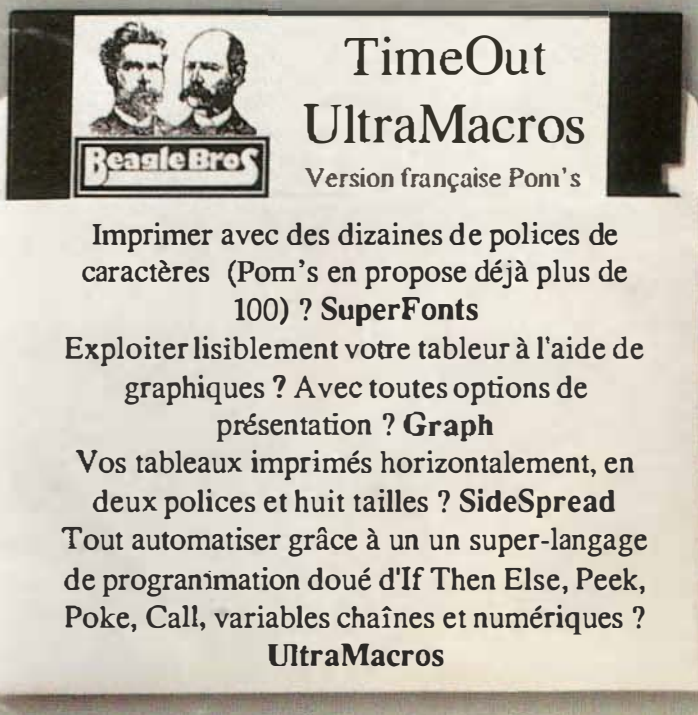
SUR LES APPLE, PROSEL...

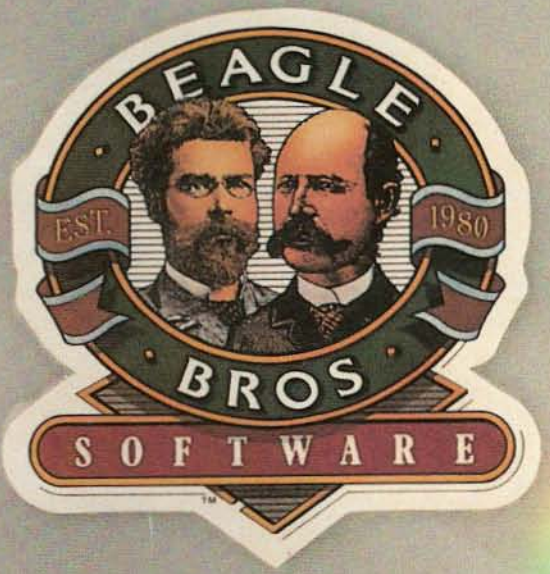
IL EST DANS LA MICRO-INFORMATIQUE APPLE DES NOMS MAGIQUES. GLEN BREDON EST DE CEUX-CI. IL EST DES PROGRAMMES QUI SIMPLIFIENT L'INFORMATIQUE, QUI OFFRENT SOUPLESSE ET LIBERTE. PROSEL EST DE CEUX-CI.

PROSEL, PROGRAM SELECTOR, EST AVANT TOUT UN SELECTEUR D'APPLICATIONS, UNE FAÇON RATIONNELLE DE TRAVAILLER. IL OFFRE UN MENU DES APPLICATIONS DE VOTRE CHOIX, MODIFIABLE A VOTRE GRE, EN FONCTION DE L'EVOLUTION DE VOS BESOINS — FONCTIONNEMENT AU CLAVIER ET PAR LA SOURIS — LANCEMENT DE PROGRAMMES PRODOS 8, PRODOS 16 ET MEME DOS 3.3... — EXTINCTION AUTOMATIQUE DE L'ECRAN APRES 15 MINUTES SANS UTILISATION —

PROSEL EST AUSSI UN ENSEMBLE D'UTILITAIRES. — RECHERCHE DE FICHIERS DANS LES DOSSIERS — DUMP DE TOUS FICHIERS EN ASCII OU TEXTE — SUPPRESSION LOGIQUE DE VOLUMES PRODOS INUTILES A UNE APPLICATION — EXECUTION PROGRAMMEE D'APPLICATIONS AVEC MINUTAGE DES SEQUENCES DE TRAVAIL — CONSTITUTION D'UNE MEMOIRE-CACHE — GESTION DES SAUVEGARDES ET RESTITUTIONS DE TOUT VOLUME PRODOS (DISQUES DURS SUR DISQUETTES) — PROGRAMME DE COPIE UNIVERSEL ET DE MANIPULATION DE FICHIERS Y COMPRIS RECUPERATION DE FICHIERS EFFACES — ANALYSE DE L'OCCUPATION DES DISQUES, ARBRE DE CATALOGUES, AFFECTATION DES BLOCS — REORGANISATION DES DISQUES — RECUPERATION DE DISQUES ABIMES — PROTECTION D'APPLICATIONS PAR MOT DE PASSE — FORMATEUR POUR LES DISQUETTES 3,5 POUCES EN DOS 3.3—

LA LIBERTE EST ACCESSIBLE : 500,00 F Y COMPRIS DOCUMENTATION FRANÇAISE, PORT 20,00 F. TARIF CARTE JOKER : 450,00 F — EDITIONS MEV/POM'S 12, RUE D'ANJOU – 78000 VERSAILLES ✆ : (1) 39 51 24 43 - COMMANDE PAR MINITEL AU : (1) 39 53 04 40





LES TIMEOUT ET APPLEWORKS 1.4

Dès aujourd'hui 9 TimeOut • SuperFonts • Graph • SideSpread • UltraMacros Data Converter • Debug • Valeurs ASCII • File Status • AWP to TXT

En français, sur l'AppleWorks 1.4 français, avec manuels français, grâce à Pom's

Éditions MEV - Pom's — 12, rue d'Anjou — 78000 Versailles — (1) 39 51 24 43 — Serveur Minitel (1) 39 53 04 40